# Historical Salmon Commerical Catch Data System of the Fisheries Research Branch, Department of Fisheries and Oceans, Pacific Region

F. Y. C. Wong

Department of Fisheries and Oceans
Fisheries Research Branch
Pacific Biological Station
Nanaimo, British Columbia V9R 5K6

March 1983

**Canadian Technical Report of Fisheries and Aquatic Sciences No. 1156**

## Canadian Technical Report of Fisheries and Aquatic Sciences

These reports contain scientific and technical information that represents an important contribution to existing knowledge but which for some reason may not be appropriate for primary scientific (i.e. *Journal*) publication. Technical Reports are directed primarily towards a worldwide audience and have an international distribution. No restriction is placed on subject matter and the series reflects the broad interests and policies of the Department of Fisheries and Oceans, namely, fisheries management, technology and development, ocean sciences, and aquatic environments relevant to Canada.

Technical Reports may be cited as full publications. The correct citation appears above the abstract of each report. Each report will be abstracted in *Aquatic Sciences and Fisheries Abstracts* and will be indexed annually in the Department's index to scientific and technical publications.

Numbers 1-456 in this series were issued as Technical Reports of the Fisheries Research Board of Canada. Numbers 457-714 were issued as Department of the Environment, Fisheries and Marine Service, Research and Development Directorate Technical Reports. Numbers 715-924 were issued as Department of Fisheries and the Environment, Fisheries and Marine Service Technical Reports. The current series name was changed with report number 925.

Details on the availability of Technical Reports in hard copy may be obtained from the issuing establishment indicated on the front cover.

## Rapport technique canadien des sciences halieutiques et aquatiques

Ces rapports contiennent des renseignements scientifiques et techniques qui constituent une contribution importante aux connaissances actuelles mais qui, pour une raison ou pour une autre, ne semblent pas appropriés pour la publication dans un journal scientifique. Il n'y a aucune restriction quant au sujet, de fait, la série reflète la vaste gamme des intérêts et des politiques du Ministère des Pêches et des Océans, notamment gestion des pêches, techniques et développement, sciences océaniques et environnements aquatiques, au Canada.

Les Rapports techniques peuvent être considérés comme des publications complètes. Le titre exact paraîtra au haut du résumé de chaque rapport, qui sera publié dans la revue *Aquatic Sciences and Fisheries Abstracts* et qui figurera dans l'index annuel des publications scientifiques et techniques du Ministère.

Les numéros 1-456 de cette série ont été publiés à titre de Rapports techniques de l'Office des recherches sur les pêcheries du Canada. Les numéros 457-714, à titre de Rapports techniques de la Direction générale de la recherche et du développement, Service des pêches et de la mer, ministère de l'Environnement. Les numéros 715-924 ont été publiés à titre de Rapports techniques du Service des pêches et de la mer, Ministère des Pêches et de l'Environnement. Le nom de la série a été modifié à partir du numéro 925.

La page couverture porte le nom de l'établissement auteur où l'on peut se procurer les rapports sous couverture cartonnée.

Canadian Technical Report of
Fisheries and Aquatic Sciences No. 1156

March 1983

# HISTORICAL SALMON COMMERCIAL CATCH DATA SYSTEM OF THE FISHERIES RESEARCH BRANCH, DEPARTMENT OF FISHERIES AND OCEANS, PACIFIC REGION

by

F. Y. C. Wong

Department of Fisheries and Oceans
Fisheries Research Branch
Pacific Biological Station
Nanaimo, British Columbia V9R 5K6

(c)Minister of Supply and Services Canada 1983

Cat. No. Fs 97-6/1156 ISSN 0706-6457

TABLE OF CONTENTS


| | Page |
|---|---|
| List of Tables | v |
| List of Figures | v |
| Abstract | vi |
| 1. Introduction | 1 |
| 2. Overview of Past System | 1 |
| 2.1 Data acquisition, transcription and verification | 1 |
| 2.2 Data storage and retrieval | 2 |
| 3. Catch Detail System | 2 |
| 3.1 Translation module | 2 |
| 3.2 Physical and logical structure | 3 |
| 3.3 Query module | 14 |
| 3.4 Report writer module | 16 |
| 4. Catch Summary System | 16 |
| 4.1 Physical and logical structure | 16 |
| 4.2 Query module | 22 |
| 4.3 Report writer module | 23 |
| 5. Possible Sources of Error | 24 |
| 6. Acknowledgments | 24 |
| 7. References | 25 |
| Appendix I. Three commonly used sales-slip forms | 26 |
| Appendix II. 72 byte sales-slip record format and definition of codes | 28 |
| Appendix III. Command files for saving and restoring CDDS files | 38 |
| Appendix IV. Sample run and result of report writer of CDDS | 39 |
| Appendix V. Sample run and result of report writer of CSDS | 41 |
| Appendix VI. Example of interfacing user program with CSDS subroutines | 53 |
| Listing A. Listing of program for translating original sales-slip records | 60 |
| Listing B. Listing of program for generating CDDS files | 62 |

TABLE OF CONTENTS


| | | Page |
|---|---|---|
| Listing C. | Listing of report writer program for CDDS ..... | 65 |
| Listing D. | Listing of program for generating CSDS files ..... | 68 |
| Listing E. | Listing of report writer program for CSDS ..... | 71 |
| Listing F. | Listing of subroutines for accessing CDDS and CSDS ..... | 84 |

## LIST OF TABLES


| | | Page |
|---|---|---|
| 1. | Conversion of COBOL alphanumeric symbols to numeric codes | 4 |
| 2. | Original and translated 72 byte record tapes | 4 |
| 3. | Time and storage comparisons of accessing the original sales-slip data and CDDS | 5 |
| 4. | CDDS tapes | 5 |


## LIST OF FIGURES


| | | Page |
|---|---|---|
| 1. | Logical structure of CDDS | 7 |
| 2. | Physical structure of CDDS - sales-slip data file | 9 |
| 3. | Physical structure of CDDS - sales-slip index file | 11 |
| 4. | Physical structure of CDDS - CFV index file | 13 |
| 5. | Logical structure of CSDS | 17 |
| 6. | Physical structure of CSDS - catch data file | 19 |
| 7. | Physical structure of CSDS - index and effort file | 21 |

## ABSTRACT

Wong, F. Y. C. 1982. Historical salmon commercial catch data system of the Fisheries Research Branch, Department of Fisheries and Oceans, Pacific Region. Can. Tech. Rep. Fish. Aquat. Sci. 1156: vi + 94 p.

A data base system was constructed to provide quick and efficient access to historical B.C. salmon commercial catch statistics. Summary catch information by year, area, time period, gear and species; number of days open; and the amount of effort are stored on-line. Detailed records of individual boat catches are stored on tape. Data can be accessed in two ways. First, a series of formatted reports can be obtained by going through a dialogue with the access programs. Second, data can be accessed by interfacing user programs with data base subroutines. This latter arrangement provides the flexibility needed for data analyses or computer modelling exercises involving the use of historical salmon catch statistics.

Key words: salmon catch statistics, data base, historical

## RÉSUMÉ

Wong, F. Y. C. 1982. Historical salmon commercial catch data system of the Fisheries Research Branch, Department of Fisheries and Oceans, Pacific Region. Can. Tech. Rep. Fish. Aquat. Sci. 1156: vi + 94 p.

Un fichier central de données a été mis au point afin de fournir un accès rapide et efficace aux statistiques historiques des prises commerciales de saumon en C.-B. Le sommaire de l'information sur les prises, soit l'année, l'endroit, la période visée, l'engin, l'espèce, le nombre de jours d'ouverture et l'effort de pêche, est stocké en ligne. Les données détaillées sur les prises de chaque bateau sont entreposées sur ruban. L'accès à l'information peut s'effectuer des deux manières suivantes: soit en obtenant une série de rapports mis en forme à l'aide d'un dialogue avec les programmes d'accès, soit en reliant les programmes de l'utilisateur avec les sous-programmes du fichier des données. Cette dernière méthode fournit la flexibilité nécessaire pour les analyses des données ou les exercices de modélisation informatisée qui comprennent l'utilisation des statistiques historiques des prises de saumon.

Mots-clés: statistiques des prises de saumon, fichier central de données, historique

# 1. INTRODUCTION

The Department of Fisheries and Oceans, Pacific Region, has kept records of all finfish and shellfish landings since 1951. From 1967 on, these records were made machine readable for processing and storage by the Block Brothers computing center. Most users access this information by consulting the B.C. Catch Statistics reports or by submitting special requests to the Block Brothers computing center where historical catch records are stored. Unfortunately, since the data system was originally designed to process sales-slips on a yearly basis, retrieval of historical data is expensive and time consuming.

The Pacific Biological Station, a research laboratory of the Department, increased its computing capability by purchasing a VAX-11 computer in 1981. As an initial step in building data bases and management models for the B.C. salmon fishery, historical salmon sales-slip data tapes (1967-present) were obtained from Block Brothers. Catch information from 1951 to 1966 were keypunched (B.C. Catch Statistics 1951-1966) and made available to all VAX-11 users. The tape-oriented "Catch Detail Data System" (CDDS) provides quick access to individual CFV records from 1967 to present. The on-line "Catch Summary Data System" (CSDS) gives summary catch information by year, area, period, gear and species from 1951 to present.

Chapter II gives an overview of the data acquisition, processing and storage of the past system. Chapter III describes the logical and physical structures of the "Catch Detail Data System" and ways of accessing it. Similarly, chapter IV describes the logical and physical structures of the "Catch Summary Data System" and the accompanying access programs and subroutines. Finally, chapter V points out some of the major errors that may exist in the data. Appendices IV and V provide sample runs and results of standard reports developed so far. Examples of interfacing user programs with data base subroutines are provided in Appendix VI.

The author of this report solicits input and suggestion from all potential users so that future versions of this data system can be modified to satisfy a wide range of user requirements.

# 2. OVERVIEW OF PAST SYSTEM

## 2.1 DATA ACQUISITION, TRANSCRIPTION AND VERIFICATION

DPA (1980) made a detailed study of the sales-slip system of the Department of Fisheries and Oceans, Pacific Region. Briefly, a sales-slip (Appendix I) is generated when catch is first sold. This usually occurs at the point of landing as most fishermen sell their fish to processing companies. The purchasing company's tallyman enters appropriate data items

and the form is signed by both the tallyman and fisherman. In most cases, the completed sales-slips are forwarded to the company's regional head offices and are also sent to D.F.O.'s Vancouver office for keypunching.

These sales-slips are first manually inspected and corrected for obvious errors such as incorrect species codes, month-day transpositions etc. They are then sorted by gear, area, period, and sent to be keypunched. The data tape is relayed to the Block Brothers IBM computer for verification. Every week, the original 100 character record is converted to an 88 character record for each species and the landed form of the fish is modified (e.g. salmon landed as dressed head-off is converted to dressed head-on by adding 10% to the weight and the price is adjusted accordingly so that the total value remains the same). The 88 character record is then checked for further errors and changed to the 72 character sales-slip record. (See D.P.A. 1980 for a detailed description of the editing procedure and format changes). Records that have both pieces and weight information are used to estimate the average weight of each species and is stored for each area, period and gear combinations. These average weights are updated as more sales-slips come in. At the end of a month, records lacking pieces information but containing weight are filled in by dividing the weight with the average weight of the particular species, area and gear concerned.

At the end of a year, all landings on a CFV are checked against gear type and capacity of the vessel. Catches exceeding vessel capacity are rejected and vessels landing less than 5 percent of their catch on a second gear are also reported. Corrections to these records are made by EDP staff after investigating original sales-slips and the CFV's licence application.

## 2.2 DATA STORAGE AND RETRIEVAL

Every year, two reports are published: the B.C. Catch Statistics and the Annual Summary of B.C. Catch Statistics. Most users access the sales-slip data by consulting these reports or by submitting special requests to Block Brothers' computing center where data tapes are stored. Unfortunately, the cost of these special requests are prohibitive for many potential users. Consequently, the Department is limited in its ability to make full usage of these valuable information to support management and research needs.

# 3. CATCH DETAIL DATA SYSTEM

## 3.1 TRANSLATION MODULE

The Block Brothers computing center maintains machine readable data files back to 1967. These files all share a common format and codes (Appendix II). For security reasons, original CFV#s are converted to

different codes so that sensitive information on specific licences or boats cannot be identified. However, the conversion algorithm remains the same every year, so that general studies on fleet movement, catch per effort, etc. can be carried out consistently from year to year.

The original data files also contained COBOL alphanumeric symbols in certain columns. Table 1 shows conversions of these alphanumeric symbols to numeric codes. A listing of the translation program is provided in Listing A. The algorithm for scrambling and unscrambling CFV codes will be provided only with written permission from appropriate authorities.

These translated data files were then stored on VAX-labelled tapes. Table 2 lists the reel #, year, record size and label of these tapes which are accessable to all F.R.B. VAX-11 users.

## 3.2 PHYSICAL AND LOGICAL STRUCTURE OF CDDS

The original and translated sales-slip data are stored as sequential files. Access to records in this form is cumbersome and time consuming. For example, to obtain catch information on a single boat in a typical year requires about 30 minutes of CPU time and up to 2 hours of elapsed time (Table 3). Such inefficiency is obviously unacceptable.

To improve the efficiency of storage and retrieval of historical sales slips information, the Catch Detail Data System (CDDS) was constructed. Original records were changed to binary codes and arranged in a tree structure with pointers pointing to major keys. Figure 1 shows the logical structure of CDDS. It is a simple tree structure nested 4 levels deep with area, period, gear and species being the major keys. A link-list of CFV# provides horizontal access to individual CFV# data. Physically, data is stored in three separate random access files. Basic data items (i.e. CFV, days fished, pieces, weight, form, price, company, day, species 2) are sorted by area, period, gear, species and CFV#; and stored in the 'Sales-slip Data File' (Fig. 2). Pointers to this file for each key (area, period, gear, species) are stored in the 'Sales-slip Index File' (Fig. 3). A third file 'CFV Index File' (Fig. 4) stores the last record location of each vessel in the 'Sales-slip Data File'.

Listing B lists the programs used for generating CDDS files. Because limited disk space is allocated for this project, CDDS data files are stored on tape using the VAX-BACKUP utility program. Table 4 lists the reel numbers and labels of these tapes (refer back to Table 3 and note increased storage efficiency). Appendix III lists the command files for saving and restoring CDDS data files from disk to tape and vice-versa.

Table 1. Conversion of COBOL alphanumeric symbols to numeric codes.

| COBOL alphanumeric | numeric |
|---|---|
| @ | 0 |
| A | 1 |
| B | 2 |
| C | 3 |
| D | 4 |
| E | 5 |
| F | 6 |
| G | 7 |
| H | 8 |
| I | 9 |

Table 2. Original and translated 72 byte record tapes.

| VAX Reel # | B.B. Tape | Year | Label | # Records |
|---|---|---|---|---|
| 25244 | BX5029 | 67 | CA67.1 | 614940 |
| 25245 | BX5034 | 67 | CA67.2 | 99096 |
| 25246 | BX5041 | 68 | CA68.1 | 616272 |
| 25347 | BX5042 | 68 | CA68.2 | 125955 |
| 25248 | BX5035 | 69 | CA69.1 | 551637 |
| 25249 | BX5043 | 70 | CA70.1 | 615384 |
| 25262 | BX5052 | 70 | CA70.2 | 105401 |
| 25251 | BX5053 | 71 | CA71.1 | 617160 |
| 25252 | BX5063 | 71 | CA71.2 | 527 |
| 25253 | BX5054 | 72 | CA72.1 | 591688 |
| 25254 | BX5055 | 73 | CA73.1 | 577166 |
| 25255 | BX5056 | 74 | CA74.1 | 495377 |
| 25256 | BX5057 | 75 | CA75.1 | 393371 |
| 25257 | BX5058 | 76 | CA76.1 | 470121 |
| 25258 | BX5081 | 77 | CA77.1 | 543830 |
| 25258 | BX5083 | 77 | CA77.1 | 56526 |
| 25259 | BX5060 | 78 | CA78.1 | 449722 |
| 25260 | BX5061 | 79 | CA79.1 | 398518 |
| 25261 | BX5062 | 80 | CA80.1 | 421527 |
| | BX5101 | 81 | | |

Table 3. Time and storage comparisons of accessing the original sales-slip data and CDDS.

| Job | Original Data Set | CDDS |
|---|---|---|
| Load from tape to disk | | |
| - CPU time | 00: 11: 40 | 00: 02: 16 |
| - Elapsed time | 00: 52: 32 | 00: 15: 42 |
| - File storage | 83207 Blks. | 26121 Blks. |
| Access single vessel data | | |
| - CPU time | 00: 41: 05 | 00: 00: 09 |
| - Elapsed time | 01: 14: 58 | 00: 00: 43 |
| Grand total | | |
| - CPU time | 00: 52: 45 | 00: 02: 25 |
| - Elapsed time | 02: 07: 30 | 00: 16: 25 |
| - File storage | 83207 Blks. | 26121 Blks. |

Table 4. CDDS tapes.

| reel # | year | label |
|---|---|---|
| 25267 | 67,68 | DB6768 |
| 25268 | 69,70 | DB6970 |
| 25269 | 71,72 | DB7172 |
| 25270 | 73,74 | DB7374 |
| 25271 | 75,76,77 | DB7577 |
| 25272 | 78,79,80 | DB7880 |

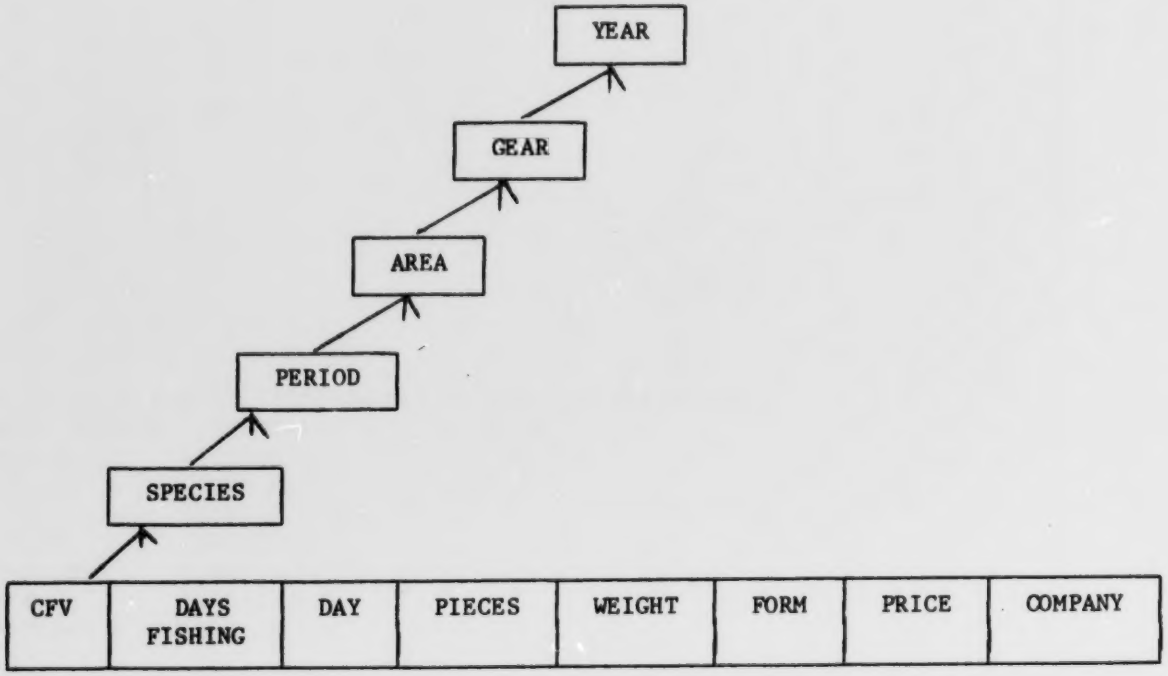
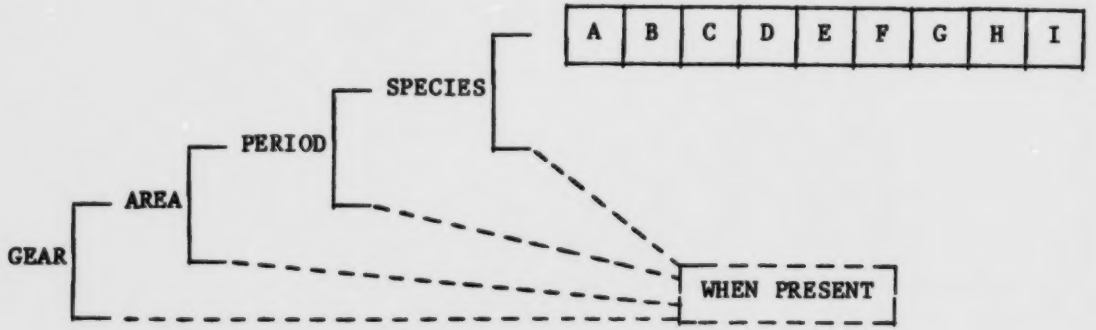



Fig. 1. Logical structure of CDDS.

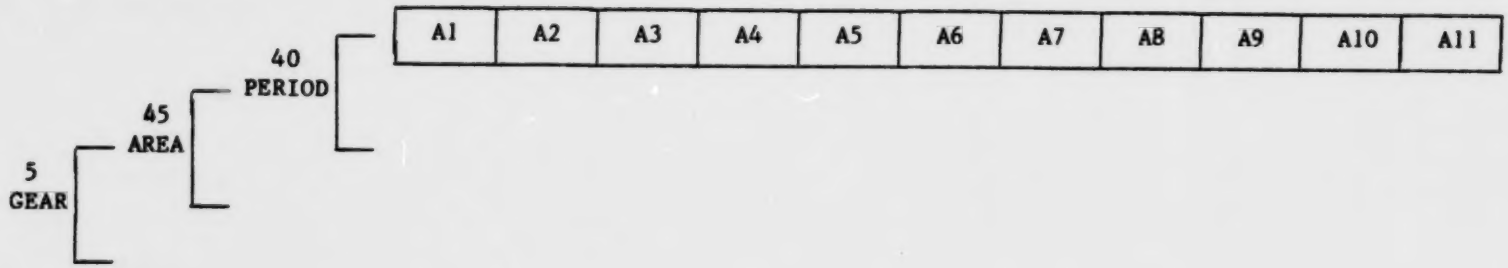



WHERE:

A = POINTER TO THE PREVIOUS RECORD OF SAME CFV (INTEGER*4)
B = DAYS FISHING (INTEGER*2)
C = PIECES (INTEGER*4)
D = WEIGHT (INTEGER*4)
E = FORM (LOGICAL*1)
F = PRICE (INTEGER*2)
G = COMPANY (LOGICAL*1)
H = DAY (LOGICAL*1)
I = SPECIES 2 (LOGICAL*1)

Note: First record of A equals to-CFV#
Last address of A and associated CFV# stored in 'CFV DATA FILE'

Fig. 2. Physical structure of CDDS - Sales-slip Data File.






WHERE:

A1 to A11 = Ending address of species 1 - 11 in 'Sales-slip Data File' (INTEGER*4)

Fig. 3. Physical structure of CDDS - Sales-slip Index File.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| RECORD # 1 | A | B | C | D | E | | | |
| RECORD # 2 | V1 | V2 | V3 | V4 | A1 | A2 | A3 | A4 |
| RECORD # 3 | " | | | | | | | |
| | " | | | | | | | |
| | " | | | | | | | |
| | " | | | | | | | |
| | " | | | | | | | |
| | " | | | | | | | |
| RECORD # n | " | | | | | | | |

WHERE:

A = Ending address of gillnet in 'CFV Index File'
B = Ending address of seine in 'CFV Index File'
C = Ending address of troll in 'CFV Index File'
D = Ending address of freezer trol in 'CFV Index File'
E = Ending address of unknown gear in 'CFV Index File'

V1-V4 = Different CFV numbers

F1-F4 = Last address of V1-V4 in 'Sales-slip Data File'

Fig. 4. Physical structure of CDDS - CFV Index File.

## 3.3 QUERY MODULE

To access CDDS it is necessary first to load the three files (Sales-slip Data File, Sales-slip Index File and CFV Index File) of a given year from tape to disk (Appendix III). Data for a given area, period, gear and species can be obtained by reading the indexes pertaining to these keys from 'Sales-slip Index File'. These indexes show where the corresponding basic data items are stored in 'Sales-slip Data File' which can be accessed directly. Similarly, data for a given vessel can be obtained easily by matching the given CFV# with those stored in 'CFV Index File'. The accompanying index shows where the last record for this vessel is stored in the 'Sales-slip Data File'. The first field of this record shows the location of the next record for the same vessel.

Several FORTRAN subroutines were written to access CDDS at various levels. The following paragraphs describe them and their passing parameters. To interface these subroutines with user programs, simply link the user object module(s) with [SALMON.CATCH] CATCH/LIB on the F.R.B. VAX-11 computer. Listings of these subroutines are provided in Listing F. Subroutines with prefix "DB" access CDDS, subroutines with prefix "CA" access CSDS (discussed in the next chapter) and subroutines with prefix "G" access both systems.

1. GFO(IYR,ISW) - GFO opens appropriate data files for processing. It must be called before other subroutines to access data of a given year.

IYR: an integer representing year (2 digits) of data wanted.
ISW: if ISW = 1, only CSDS files are opened for read-only access. Logical units 11 and 12 are assigned to the data and index files respectively. If ISW <>1, CDDS files must be loaded from tape to disk before this subroutine is called. Logical units 11 to 16 are assigned to CDDS and CSDS files accordingly.

Note: GFO closes files opened by previous calls to GFO so that no more than one year's files are opened concurrently.

2. DBCFV(IGR,ICFV,IDX) - Given a CFV# and gear code, DBCFV returns location of the last record for the given CFV# stored in 'Sales-slip Data File'.

IGR: CDDS gear code
ICFV: CDDS scrambled CFV#
IDX: integer to receive location of the last record of ICFV stored in "Sales-slip Data File".

Note: CDDS files must be loaded from tape to disk. GFO must be called before this subroutine.

3. DBID1(IGR); DBID2(IDX,IAR,IPR,ISP) - Given record number of 'Sales-slip Data File', DBID2 returns the area code, period code and species code of that record.

IGR: CDDS gear code
IDX: record number of 'Sales-slip Data File' CDDS
IAR: integer to receive area code
IPR: integer to receive period code
ISP: integer to receive species code

Note: CDDS files must be loaded from tape to disk. GFO must be called before DBID1 and DBID2. DBID1 must be called at least once for each gear before DBID2. DBID1 nullifies previous calls to itself so that DBID2 calls should follow immediate after calls to DBID1.

4. DBCFVD(IGR,IYR,ICFV,ISW,C,DF): Given gear code, year and scrambled CFV code, DBCFVD returns catch by species, area and period; and the number of days fished by the given CFV#.

IGR: CDDS gear code
IYR: year (2 digits)
ISW: returns catch in pieces if ISW=1, otherwise catch is in weight
C: 11X40X45 integer*4 array to recieve catch by 11 species, 40 periods and 45 areas
DF: 40x45 integer*4 array to receive days fished by 40 periods and 45 areas

Note: CDDS files must be loaded from tape to disk. GFO must be called before this subroutine.

5. DBIDX(IGR, IAR, IPR, ISP, IS, IE) - Given gear, area, period and species codes, DBIDX returns starting and ending record number of data stored in 'Sales-slip Data File' for the given keys.

IGR: CDDS gear code
IAR: CDDS area code
IPR: CDDS period code
ISP: CDDS species code
IS: integer variable to receive starting record number
IE: integer variable to receive ending record number

Note: CDDS files must be loaded from tape to disk. GFO must be called before this subroutine.

6. DBACC(IDX, INCFV, IDF, IPS, IWT, IFM, IPI, KCO, IDY, IS2) - Given record number, DBACC returns the 9 basic data items of CDDS.

IDX: record number of data wanted (see DBIDX)
INCFV: integer*4 variable to receive record location of previous record of same CFV#
IDF: integer*2 variable to receive days fished data
IPS: integer*4 variable to receive pieces data
IWT: integer*4 variable to receive weight data

IFM: logical*1 variable to receive form code
IPI: integer*2 variable to receive price data
KCO: logical*1 variable to receive company code
IDY: logical*1 variable to receive day code
IS2: logical*1 to receive species 2 code

Note: CDDS files must be loaded from tape to disk. GFO must be called before this routine. See DBIDX for obtaining record number. See Appendix II for data item descriptions.

### 3.4 REPORT WRITER MODULE

Only one standard output report has been developed so far to access CDDS. It provides catch (in pieces or in weight) by gear, area, period and species of individual scrambled CFV#s in a given year. Appendix IV shows a sample run and result. A listing of the program is provided in Listing C. Suggestions for additional report formats and corrections to the present report format will be gratefully received.

## 4. CATCH SUMMARY DATA SYSTEM

### 4.1 PHYSICAL AND LOGICAL STRUCTURE

To provide quick access to data frequently used for management and biological research, a subset of CDDS is stored on-line (Catch Summary Data System, "CSDS"). Its logical structure is shown in Figure 5. It is a simple tree structure nested 3 levels deep with area, period and gear being major keys. Physically, data are stored in 2 separate random access files. Catches in pieces and weight are sorted by species, gear, period, area; and then stored in 'Catch Date File' (Fig. 6). Pointers to 'Catch Data File' for each key (area, period, gear) are stored in 'Index and Effort File'. Other data items such as effort, days opened and days fished are also stored in the 'Index and Effort File' (Fig. 7).

Catch in pieces or weight for a given area, period, gear, and species can be obtained easily by reading the index associated with the given keys from the 'Index and Effort File' which shows where the data items are stored in the 'Catch Data File'. Effort, days open and days fished data can be obtained directly from the 'Index and Effort File'.

Listing D lists the program used for generating CSDS files.

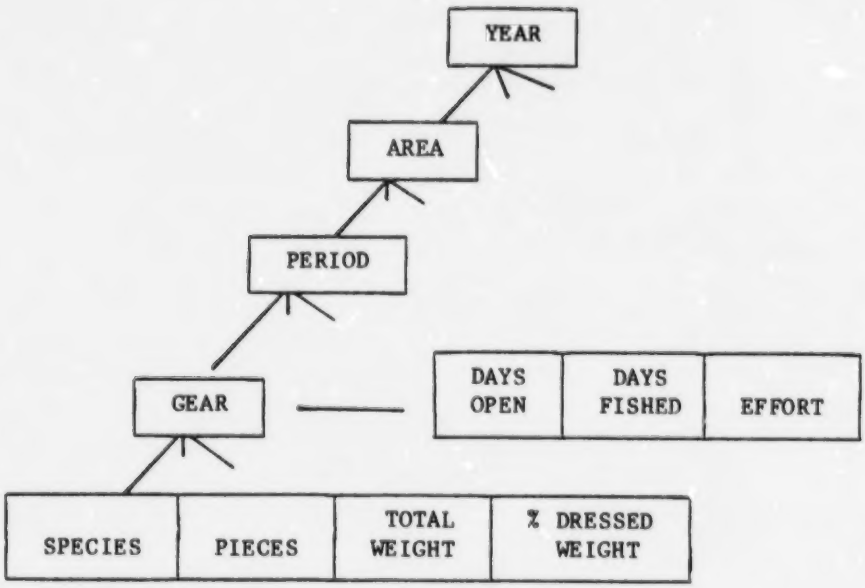
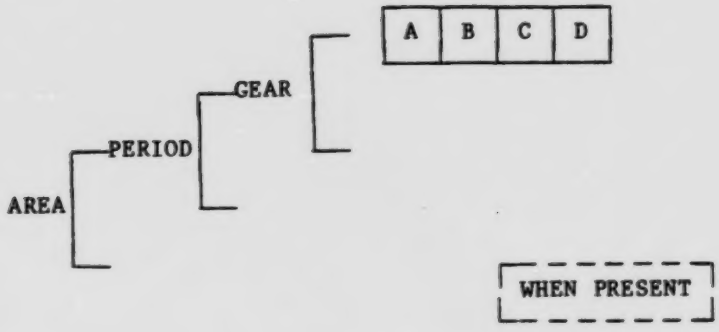



Fig. 5. Logical structure of CSDS.




WHERE:

A = SPECIES CODE (LOGICAL *1)

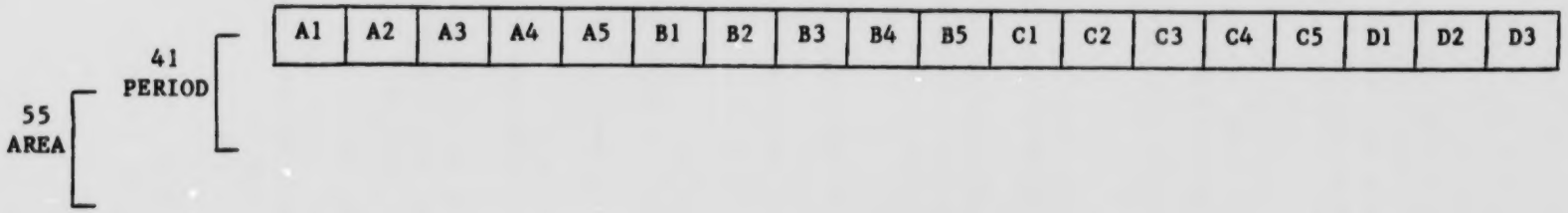
B = PIECES (INTEGER *4)

C = TOTAL WEIGHT (INTEGER *4)

D = PERCENT DRESS WEIGHT (LOGICAL *1)

NOTE: 6 bytes free space per record for further expansion

Fig. 6. Physical structure of CSDS - catch data file




WHERE:

| | | |
|---|---|---|
| A1 - A5 | = Ending record number by gear in "catch data file" | (INTEGER*4) |
| B1 - B5 | = Number of boat by gear | (INTEGER*2) |
| C1 - C5 | = Total days fished by gear | (INTEGER*2) |
| D1 - D3 | = Number of days open by gear | (LOGICAL*1) |

NOTE: A5, B5, C5 deal with unknown gear and are usually empty. 1 byte free space per record.

Fig. 7. Physical structure of CSDS - Index and effort file.

## 4.2 QUERY MODULE

Three FORTRAN subroutines were written to access CSDS. The following paragraphs describe these subroutines and their passing parameters, To interface these subroutines with user defined programs, simply link the user object module(s) with [SALMON.CATCH] CATCH/LIB on the F.R.B. VAX-11 computer (see Appendix VI for a sample user program). Listings of these subroutines are provided in Listing F.

1. CAA1(IAR, IPR, ISW, C, TGR, TDF, DO) - Given area and period, CAA1 returns catch by species by gear, number of boats by gear, days fished by gear and days open by gear.

IAR: CSDS area code
IPR: CSDS period code
ISW: returns catch in pieces if ISW=1, otherwise catch is in weight
C: 11X5 integer*4 array to receive catch by 11 species and 5 gears
TGR: 5X1 integer*4 array to receive number of boats by 5 gears
TDF: 5X1 integer*4 array to receive total days fished by 5 gears
DO: 3X1 integer*4 array to receive days open by gears 1-3. (see CSDS gear code)

Note: GFO must be called before this routine to open files appropriate for the year concernced. CAA1 calls two lower level routines to access the index file and data file. See CAGR and CAREC for more detailed description of these subroutines.

2. CAGR(IAR, IPR, ID, TGR, TDF, DO) - Given area and period, CAGR returns the last record location of catch by gear stored in 'Catch Data File', number of boats by gear, days fished by gear and days open by gear.

IAR: CSDS area code
IPR: CSDS period code
ID: 5X1 integer*4 array to receive the ending record location of catch data by 5 gears
TGR: 5X1 integer*4 array to receive number of boats by 5 gears
TDF: 5X1 integer*4 array to receive total days fished by 5 gears
DO: 3X1 integer*4 array to receive total days open by gears 1-3 (see CSDS gear code)

Note: CAGR provides quick access to days open, days fished and gear data. If only catch data is required, use CAA1 instead. GFO must be called before this subroutine.

3. CAREC(IAR, IPR, IGR, ID, CA) - Given area, period, gear and record location, CAREC returns the basic data items of 'Catch Data File' by species.

IAR: CSDS area code
IPR: CSDS period code
IGR: CSDS gear code
ID: 5X1 integer*4 array containing the ending record location by 5 gears (obtained from CAGR)
CA: 11X3 integer*4 array to receive pieces caught, total weight and percent dressed weight by 11 species.

Note: GFO and CAGR must be called before this subroutine.

## 4.3 REPORT WRITER MODULE

Four standard output reports have been developed and implemented to access CSDS. Parameters such as year(s) and area(s) of data wanted are specified by the user through a dialogue with the main program (sample runs and results are provided in Appendix V). These reports can be previewed on a video terminal before printing so that unnecessarily large or unwanted printouts can be avoided. Listing E lists the main program and subroutines used to generate these reports.

Several FORTRAN subroutines used in the above program can be used to interface with user programs. The following paragraphs describe their functions and passing parameters.

1. GLIB1(AR, IAR, IW) - Given DFO statistical area, GLIB1 returns CSDS area code and vice-versa.

AR: character*7 string containing (or to receive) DFO statistical area code
IAR: integer*4 variable containing (or to receive) CSDS area code
IW: if IW=1, CSDS area code is given to receive DFO statistical area code, otherwise DFO statistical area code is given to receive CSDS area code

2. GLIB2 (YR, IPR, MO, IDAY) - Given year and period, GLIB2 returns the ending date of that period.

YR: character*2 string containing year concerned
IPR: CSDS period code
MO: integer variable to receive numerical month
IDAY: integer variable to receive day of the month

## 5. POSSIBLE SOURCES OF ERROR

DPA (1980) conducted an extensive interview with users of the catch statistics system to discuss major issues and problems. The major points are reviewed in this chapter to make users aware of the potential pitfalls in using these data for analysis.

Concerns have been voiced about possible increases in unreported landings as cash-buyers and direct sales to the public becomes more popular among fishermen.

The 72-byte records used in this project have gone through various editing and verification processes. Unfortunately, original sales-slips were not given unique identification codes so that it is very difficult to verify data records with original sales-slips when the need arises.

Some original (or missing) data items are changed (or filled in) without adequate documentation or rationale behind the changes. For example, missing "pieces" are filled in by dividing recorded weight with the average weight estimated from other sales-slips. More recently, catches landed in an area (or period) that was not open are changed by some ad hoc process (e.g. by changing the period code to the previous period if the recorded area was open in the previous period; or change the area code to an adjacent area that was open in the recorded period). Days fishing information for gill net and seine is lost during data entry because key-punch personnel have been instructed to change it to a "1" to indicate the number of deliveries. Generally, there is no way of knowing which record and/or data item has been changes (or filled in) from the 72-byte records.

The present system cannot handle landings caught from multiple areas and/or multiple periods (e.g. landings by freezer trollers, stored fish that are sold in the winter, etc.).

All landing forms are converted to round weights but the conversion factors have not been verified lately.

The above deficiencies point to an urgent need for redesigning the catch statistics system. Specifications for a new system have been prepared (DPA 1982). Implementation of the new system would certainly provide more accurate and timely information for all users within the Department and industry in the near future.

## 6. ACKNOWLEDGMENTS

I would like to thank J. Schnute for providing helpful suggestions, encouragement and editorial comments. Special thanks are due to S. Henderson for providing the tapes essential to this project. The staff at PBS computing

center were helpful in all aspects of computing problems. R. Lawrence performed most of the work on error checking and data coding.

## 7. REFERENCES

Anon. 1951-1980. British Columbia Catch Statistics. Dept. of Fisheries, Pacific Region.

DPA 1980. Review of Pacific Region Statistical System, Final Report. DPA Consulting Ltd., Vancouver, B.C. 82 pp.

DPA 1982. Specifications for a Commercial Catch Statistics System, Draft Report. DPA Consulting Ltd., Vancouver, B.C.

APPENDIX I. Commonly used sales-slip forms

---

**BOOK FISH**
(SHORT FORM)

COMPANY NAME

FISHERMAN'S NAME & ADDRESS

AREA_____

C.F.V. NO.

DAYS FISHING_____

S.I. #

_____ / _____ / _____
MONTH DAY YEAR

BOAT NAME

DELIVERED TO:

GEAR: GILLNET ☐ SEINE ☐
TROLL ICE ☐ FREEZER ☐

OTHER_____

| PCS. | WEIGHT | CODE | SPECIES | PRICE | VALUE | |
|---|---|---|---|---|---|---|
| | | | Sockeye Rd. | | | |
| | | | Cohoe Rd. | | | |
| | | | " | | | |
| | | | Pinks Rd. | | | |
| | | | " | | | |
| | | | Chums Rd. | | | |
| | | | " | | | |
| | | | Red Spring Rd. | | | |
| | | | " " | | | |
| | | | " " | | | |
| | | | " " | | | |
| | | | Whitespring Rd. | | | |
| | | | " " | | | |
| | | | Steelheads Rd. | | | |
| | | | Jacks | | | |
| | | | Salmon eggs | | | |
| | | | | | | |
| | | | | | | |

I CERTIFY THAT THE ABOVE INFORMATION IS COMPLETE AND CORRECT.

Y0455

TALLYMAN

BFS 25-2 SMITH GRANT MANN LIMITED BUSINESS FORMS & SYSTEMS DIVISION

## APPENDIX I. (cont'd)

NOT NEGOTIABLE

**CASH FISH**
(OPPOSITE SHOWN)

FIRM'S COPY 1

COMPANY NAME

OWNER'S NAME & ADDRESS

AREA OF CATCH _____

D.F.V. NO.

DAYS FISHING _____

S.I. #

DAY MONTH YEAR

BOAT NAME

DELIVERED TO:

GEAR GILLNET ☐ SEINE ☐
TROLL ICE ☐ FREEZER ☐
OTHER _____

| SKIPPER'S NAME | SIN # | % |
|---|---|---|
| CREW NAME | SIN # | % |
| CREW NAME | SIN # | % |
| CREW NAME | SIN # | % |
| CREW NAME | SIN # | % |
| CREW NAME | SIN # | % |
| CREW NAME | SIN # | % |

| PCS | WEIGHT | CODE | SPECIES | PRICE | VALUE |
|---|---|---|---|---|---|
| | | | DR. Large Red | | |
| | | | DR. Med. Red | | |
| | | | DR. Small Red | | |
| | | | | | |
| | | | | | |
| | | | DR. White | | |
| | | | DR. White | | |
| | | | DR. Sockeye | | |
| | | | DR. Coho | | |
| | | | DR. Coho | | |
| | | | | | |
| | | | DR. Pinks | | |
| | | | DR. Chums | | |
| | | | DR. Steelhead | | |
| | | | | | |
| | | | Round Red | | |
| | | | Round Red | | |
| | | | Round White | | |
| | | | Round Jacks | | |
| | | | Round Sockeye | | |
| | | | Round Coho | | |
| | | | Round Pink | | |
| | | | Round Chum | | |
| ■ | | | Salmon Eggs | | |
| ★ | | | Halibut Large | | |
| ★ | | | Halibut Medium | | |
| ■ | | | Ling Cod | | |

G 2176

TOTAL CATCH VALUE
U.I.C. DEDUCTIONS
CASH PAYMENT

I certify that the above information is complete and correct.

FISHERMAN'S SIGNATURE

VALLYMAN

NOT NEGOTIABLE

**BOOK FISH**
(LONG FORM)

COMPANY NAME

OWNER'S NAME & ADDRESS

AREA OF CATCH _____

D.F.V. NO.

DAYS FISHING _____

S.I. #

DAY MONTH YEAR

BOAT NAME

DELIVERED TO:

GEAR GILLNET ☐ SEINE ☐
TROLL ICE ☐ FREEZER ☐
OTHER _____

| SKIPPER'S NAME | SIN # | |
|---|---|---|
| CREW NAME | SIN # | |
| CREW NAME | SIN # | |
| CREW NAME | SIN # | |
| CREW NAME | SIN # | |
| CREW NAME | SIN # | |
| CREW NAME | SIN # | |

| PCS | WEIGHT | CODE | SPECIES | PRICE | VALUE |
|---|---|---|---|---|---|
| | | | Red Spring Dr L | | |
| | | | " " " M | | |
| | | | " " " S | | |
| | | | | | |
| | | | | | |
| | | | White Spring Dr L | | |
| | | | " " " M | | |
| | | | " " " S | | |
| | | | | | |
| | | | Sockeye Dr | | |
| | | | Coho Dr | | |
| | | | Pink Dr | | |
| | | | Chum Dr | | |
| | | | | | |
| | | | Sockeye Round | | |
| | | | Coho Round | | |
| | | | Pink Round | | |
| | | | Chum Round | | |
| ■ | | | Salmon Eggs | | |
| ★ | | | Halibut - Large | | |
| ★ | | | Halibut - Medium | | |
| ■ | | | Ling Cod D.H. Oil | | |

TOTAL VALUE OF FISH
LESS CASH RECEIVED
CREDIT TO ACCOUNT

I CERTIFY THAT THE ABOVE INFORMATION IS COMPLETE AND CORRECT.

VALLYMAN

G 4952

BASIS

FISHERMAN'S SIGNATURE

Appendix II. 72 Byte sales-slip record format and definition of codes.

Sales-slip Data Format (1967-1981)

| Column | Name | Description |
|---|---|---|
| 1- 2 | year | year |
| 3- 4 | gear | (see gear code) |
| 5- 6 | day | calender day |
| 7- 9 | period | (see period code) |
| 10-12 | area 1 | (see area code) |
| 13-14 | sub-area | N/A |
| 15 | district | (see district code) |
| 16-20 | day fishing | days fished if trol, otherwise # of deliveries |
| 21-22 | company | (see company code) |
| 23-24 | plant | N/A |
| 25-26 | transfer | N/A |
| 27-31 | CFV | CFV license # |
| 32-33 | species 1 | (see species code) |
| 34-40 | pieces | # landed |
| 41-48 | weight | weight in pound |
| 49 | form | (see form code) |
| 50-56 | amount | total amount paid in cent |
| 57-58 | species 2 | (see species code) |
| 59-61 | area 2 | N/A |
| 62 | by-pass | N/A |
| 63-64 | H sets | N/A |
| 65-72 | price | price in cent/10 |

N/A - not applicable to salmon.

Appendix II (cont'd)

| Gear Code | | |
|---|---|---|
| CDDS CSDS | SALES-SLIP | DESCRIPTION |
| 1 | 10 | gillnet |
| 2 | 20 | seine |
| 3 | 30 | troll |
| 4 | 31 | freezer troll |
| 5 | other | unknown |

APPENDIX II (cont'd)

**PERIOD CODE**

| CDDS CSDS | SALES-SLIP | DESCRIPTION |
|---|---|---|
| 1 | 031 | Each period has 7 days and each month has 4 periods except that month 04,07, and 10 has 5 periods. Period 40 in CDDS and CSDS contain all periods from December to February inclusive. |
| 2 | 032 | |
| 3 | 033 | |
| 4 | 034 | |
| 5 | 041 | |
| 6 | 042 | |
| 7 | 043 | |
| 8 | 044 | The following table shows the year and ending date of period 051. The ending date for other periods can be derived from this baseline. |
| 9 | 045 | |
| 10 | 051 | |
| 11 | 052 | |
| 12 | 053 | |
| 13 | 054 | |
| 14 | 061 | |
| 15 | 062 | |
| 16 | 063 | |
| 17 | 064 | |
| 18 | 071 | |
| 19 | 072 | |
| 20 | 073 | |
| 21 | 074 | |
| 22 | 075 | |
| 23 | 081 | |
| 24 | 082 | |
| 25 | 083 | |
| 26 | 084 | |
| 27 | 091 | |
| 28 | 092 | |
| 29 | 093 | |
| 30 | 094 | |
| 31 | 101 | |
| 32 | 102 | |
| 33 | 103 | |
| 34 | 104 | |
| 35 | 105 | |
| 36 | 111 | |
| 37 | 112 | |
| 38 | 113 | |
| 39 | 114 | |
| 40 | Other | |
| 41 | Total (all periods) | |

| Year | Ending date | Year | Ending date |
|---|---|---|---|
| 1951 | May 12 | 1966 | May 7 |
| 1952 | May 10 | 1967 | May 13 |
| 1953 | May 9 | 1968 | May 11 |
| 1954 | May 8 | 1969 | May 10 |
| 1955 | May 7 | 1970 | May 9 |
| 1956 | May 12 | 1971 | May 8 |
| 1957 | May 11 | 1972 | May 6 |
| 1958 | May 10 | 1973 | May 5 |
| 1959 | May 9 | 1974 | May 11 |
| 1960 | May 7 | 1975 | May 10 |
| 1961 | May 13 | 1976 | May 8 |
| 1962 | May 12 | 1977 | May 7 |
| 1963 | May 11 | 1978 | May 13 |
| 1964 | May 9 | 1979 | May 12 |
| 1965 | May 8 | 1980 | May 10 |

APPENDIX II (cont'd)

| AREA CODE | | |
|---|---|---|
| CDDS CSDS | SALES-SLIP | STATISTICAL AREA |
| 1 | 010 | 1 |
| 2 | 021 | 2E |
| 3 | 022 | 2W |
| 4 | 031 | 3X |
| 5 | 032 | 3Y |
| 6 | 033 | 3Z |
| 7 | 040 | 4 |
| 8 | 056 | 5/6 |
| 9 | 051 | 5/1 |
| 10 | 052 | 5/2 |
| 11 | 053 | 5/3 |
| 12 | 054 | 5/4 |
| 13 | 060 | 6 |
| 14 | 070 | 7 |
| 15 | 080 | 8 |
| 16 | 090 | 9 |
| 17 | 100 | 10 |
| 18 | 110 | 11 |
| 19 | 120 | 12 |
| 20 | 130 | 13 |
| 21 | 140 | 14 |
| 22 | 150 | 15 |
| 23 | 160 | 16 |
| 24 | 170 | 17 |
| 25 | 180 | 18 |
| 26 | 190 | 19 |
| 27 | 200 | 20 |
| 28 | 210 | 21 |
| 29 | 220 | 22 |
| 30 | 230 | 23 |
| 31 | 240 | 24 |
| 32 | 250 | 25 |
| 33 | 260 | 26 |
| 34 | 270 | 27 |
| 35 | 280 | 28 |
| 36 | 791 | 29AB |
| 37 | 792 | 29C |
| 38 | 793 | 29D |
| 39 | 794 | 29E |
| 40 | 300 | 30 |
| 41 | 500-600 | C |
| 42 | 400 | Alaska |
| 43 | 441 | Taku |
| 44 | 442 | Stikine |

APPENDIX II (cont'd)

| AREA CODE (cont'd) | | |
|---|---|---|
| CDDS CSDS | SALES-SLIP | STATISTICAL AREA |
| 45 | other | Unknown |
| 46 | - | 2AE*** |
| 47 | - | 2BE*** |
| 48 | - | 2AW*** |
| 49 | - | 2BW*** |
| 50 | - | 3** |
| 51 | - | 5** |
| 52 | - | D1* |
| 53 | - | D2* |
| 54 | - | D3* |
| 55 | - | BC |

*D1 includes stat. area 28, 29AB, 29C, 29D, 29E
*D2 includes stat. area 1-10, 30
*D3 includes stat. area 11-29, C
*BC includes stat. area 1-30, C
** total of subareas
*** subarea code used from 1956-1966

APPENDIX II (cont'd)

| DISTRICT CODE | |
|---|---|
| SALES-SLIP CSDS, CDDS | INCLUDE STATISTICAL AREAS: |
| 1 | 28, 29AB, 29C, 29D, 29E |
| 2 | 1 - 10, 30, Alaska |
| 3 | 11 - 27, C. |

APPENDIX II (cont'd)

| COMPANY CODE | | |
|---|---|---|
| CDDS | SALES-SLIP | DESCRIPTION |
| 1 | 67 | Anderson Cove oysters |
| 2 | 33 | Atlin |
| 3 | 50 | Babcock |
| 4 | 54 | B.C.F. Ind. Co-op. |
| 5 | 20 | B.C. Packers/Nelson |
| 6 | 51 | Billingsgate |
| 7 | 72 | Bingham |
| 8 | 53 | Booth |
| 9 | 52 | Campbell River |
| 10 | 30 | Canadian Fish |
| 11 | 32 | Canoe Pass Co-op. |
| 12 | 56 | Carlyle |
| 13 | 57 | Cassiar |
| 14 | 59 | Clausen |
| 15 | 88 | Coast Shellfish |
| 16 | 11 | Coltran |
| 17 | 61 | Dennis Shellfish |
| 18 | 58 | Eyford |
| 19 | 80 | Francis Millerd |
| 20 | 86 | Harbor Fisheries |
| 21 | 60 | J. H. Todd |
| 22 | 63 | Johnston |
| 23 | 64 | Lion's Gate |
| 24 | 92 | Long Beach Shell |
| 25 | 65 | McCallum Sales |
| 26 | 66 | D. McMillan |
| 27 | 94 | J. S. McMillan |
| 28 | 21 | Norm McLellan |
| 29 | 68 | Millbanke |
| 30 | 69 | Murray Fish |
| 31 | 74 | Norpae |
| 32 | 73 | North Star |
| 33 | 75 | Oakland |
| 34 | 76 | Ocean Fish |
| 35 | 62 | Ostergaard |
| 36 | 77 | Pacific Seafoods |
| 37 | 40 | Portuguese Joe |
| 38 | 70 | Pr. Rupert Co-op. |
| 39 | 82 | Q.C. Fisheries |
| 40 | 83 | Reliance |
| 41 | 96 | Royal |
| 42 | 79 | Satellite |
| 43 | 84 | Seafood Products |
| 44 | 91 | Seven-Seas |
| 45 | 85 | Shearer |

## APPENDIX II (cont'd)

| COMPANY CODE (cont'd) | | |
|---|---|---|
| CDDS | SALES-SLIP | DESCRIPTION |
| 46 | 87 | Sooke Seafoods |
| 47 | 93 | St. Jean's |
| 48 | 97 | Tofino Fisheries |
| 49 | 78 | Tofino Packing |
| 50 | 89 | Ucluelet Fish |
| 51 | 90 | Vancouver Shell |
| 52 | 98 | Direct Sales |
| 53 | 99 | Others |

APPENDIX II (cont'd)

| FORM CODE | | |
|---|---|---|
| CDDS CSDS | SALES-SLIP | DESCRIPTION |
| 1 | 1 | Round - pcs. & wt. |
| 2 | 2 | Round - wt. only |
| 3 | 3 | Dressed H/on - pcs. & wt. |
| 4 | 4 | Dressed H/on - wt. only |
| 5 | 5 | Pieces only |
| 6 | 6 | Dressed H/off |

APPENDIX II (cont'd)

SPECIES CODE

IF CAUGHT BY TROLL GEAR (SPECIES 2)

| CDDS CSDS | SALES-SLIP | DESCRIPTION |
|---|---|---|
| 1 | 10 | Red spring L (over 12 lbs.) |
| 2 | 11 | Red spring M (8 - 12 lbs.) |
| 3 | 12 | Red spring S (1 - 7 lbs.) |
| 4 | 13 | Red spring #2 (under 7 lbs.) |
| 5 | 14 | White spring |
| 6 | 15 | Sockeye |
| 7 | 16 | Coho |
| 8 | 17 | Pink |
| 9 | 18 | Chum |
| 10 | 19 | Steelhead |
| 11 | other | Unknown (1967-81)<br>Mixed spring (1951-66) |

IF CAUGHT BY GILLNET OR SEINE (SPECIES 1)

| CDDS CSDS | SALES-SLIP | DESCRIPTION |
|---|---|---|
| 1 | 10 | Red spring L. |
| 2 | 11 | Red spring M. |
| 3 | 12 | Jacks (under 5 lbs.) |
| 4 | - | - |
| 5 | 14 | White spring |
| 6 | 15 | Sockeye |
| 7 | 16 | Coho |
| 8 | 17 | Pink |
| 9 | 18 | Chum |
| 10 | 19 | Steelhead |
| 11 | other | Unknown |

Note: species 5 - all spring (1951-1962)
species 1 - red spring (L. + M.) (1963-1966)
species 5 troll - mixed troll, all spring (1963-1966)

APPENDIX III. Command files for saving and restoring CDDS files.

SAVE FILES ON DISK TO TAPE

```
$ON WARNING THEN EXIT
$ALLO MTAO:
$ON WARNING THEN GOTO FINA
$REQUEST/REPLY " 25267 RING IN "
$SET DEF DRA1:[SALMON.REC]
$INIT/DENS=1600/OVERRIDE=(E,A) MTAO:DB6768
$MOUNT/FOREIGN MTAO:DB6768
$BACKUP/LOG/VERIFY [SALMON.REC]*67*.*,*68*.* MTAO:DB6768
$FINA:
$ON WARNING THEN GOTO FINB
$DISMOUNT MTAO:
$FINB:
$DEALLO MTAO:
```

RESTORE FILES FROM TAPE TO DISK

```
$ON WARNING THEN EXIT
$ALLO MTAO:
$ON WARNING THEN GOTO FINA
$REQUEST/REPLY " 25267 RING OUT "
$SET DEF DRA1:  [SALMON.CATCH]
$MOUNT/FOREIGN MTAO:
$BACKUP/LOG MTAO:DB6768/SELECT=[SALMON.REC]*68*.*-
 [SALMON.CATCH]*/NEW VERSION
$FINA:
$ON WARNING THEN GOTO FINB
$DISMOUNT MTAO:
$FINB:
$DEALLOCATE MTAO:
$STOP
```

APPENDIX IV. Sample run and result of report writer of CDDS.

```
$TYPE T25272.COM

$ON WARNING THEN EXIT
$ALLO MTAO:
$ON WARNING THEN GOTO FINA
$REQUEST/REPLY '25272 RING OUT'
$SET DEF DRA1:[SALMON.CATCH]
$MOUNT/FOREIGN MTAO:
$BACKUP/LOG MTAO:DB7880/SELECT=[SALMON.REC]*80*.* [SALMON.CATCH]*/NEW_VERSION
$FINA:
$ON WARNING THEN GOTO FINB
$DISMOUNT MTAO:
$FINB:
$DEALLOCATE MTAO:
$STOP

$SQT T25272.COM

JOB 6942 ENTERED ON QUEUE SYS$TAPE

(ALLOW TIME FOR TAPE JOB TO FINISH)

$RUN [SALMON.CATCH]FLEET

PROGRAM TO PRINT OUT CATCH BY PERIOD, AREA, SPECIES FOR GIVEN CFV#

ENTER YEAR(67<=YEAR<=80)

80

ENTER SCRAMBLED CFV#

47418

(TAKES ABOUT 30 SECONDS TO FINISH)

OUTPUT STORED IN FILE 'SCRAT,DAT'

TYPE 'PRINT SCRAT.DAT' TO GET PRINTOUT

TRY AGAIN (Y/N)?

N

FORTRAN STOP
```

APPENDIX IV. (cont'd)

1980 CATCH STATISTICS OF SCRAMBLED CFV#: 47418( SEINE )

| PERIOD | AREA | SPRING | SOCKEYE | COHO | PINK | CHUM | STEELHEAD | TOTAL | DAYS FISHED |
|---|---|---|---|---|---|---|---|---|---|
| 3- 8 | | | | | | | | | |
| 3-15 | | | | | | | | | |
| 3-22 | | | | | | | | | |
| 3-29 | | | | | | | | | |
| 4- 5 | | | | | | | | | |
| 4-12 | | | | | | | | | |
| 4-19 | | | | | | | | | |
| 4-26 | | | | | | | | | |
| 5- 3 | | | | | | | | | |
| 5-10 | | | | | | | | | |
| 5-17 | | | | | | | | | |
| 5-24 | | | | | | | | | |
| 5-31 | | | | | | | | | |
| 6- 7 | 23 | 6 | 426 | 0 | 0 | 0 | 0 | 432 | 1 |
| 6-14 | | | | | | | | | |
| 6-21 | 23 | 3 | 110 | 0 | 0 | 0 | 0 | 113 | 6 |
| 6-28 | 23 | 13 | 429 | 0 | 0 | 0 | 0 | 442 | 3 |
| 7- 5 | 23 | 0 | 96 | 0 | 0 | 0 | 0 | 96 | 1 |
| 7-12 | 23 | 1 | 612 | 0 | 0 | 0 | 0 | 613 | 2 |
| 7-19 | 23 | 0 | 814 | 0 | 0 | 0 | 0 | 814 | 2 |
| 7-26 | 8 | 42 | 874 | 0 | 2031 | 90 | 0 | 3037 | 2 |
| 8- 2 | 8 | 46 | 71 | 23 | 906 | 750 | 0 | 1796 | 1 |
| 8- 2 | 12 | 10 | 448 | 3 | 347 | 4 | 0 | 812 | 1 |
| 8- 9 | | | | | | | | | |
| 8-16 | 1 | 3 | 1840 | 1 | 238 | 21 | 0 | 2103 | 1 |
| 8-16 | 12 | 36 | 1933 | 7 | 449 | 31 | 0 | 2456 | 3 |
| 8-23 | 12 | 18 | 38 | 12 | 35 | 1 | 0 | 104 | 1 |
| 8-30 | 8 | 178 | 0 | 76 | 358 | 86 | 0 | 696 | 1 |
| 9- 6 | 12 | 37 | 42 | 138 | 58 | 123 | 0 | 398 | 1 |
| 9-13 | 12 | 0 | 0 | 34 | 2 | 69 | 0 | 105 | 1 |
| 9-20 | 12 | 2 | 0 | 31 | 0 | 118 | 0 | 151 | 1 |
| 9-27 | 12 | 0 | 0 | 0 | 0 | 613 | 0 | 613 | 1 |
| 10- 4 | 12 | 23 | 0 | 17 | 0 | 130 | 0 | 170 | 1 |
| 10-11 | 26 | 4 | 0 | 24 | 0 | 1765 | 0 | 1793 | 2 |
| 10-18 | 12 | 14 | 0 | 19 | 0 | 951 | 0 | 984 | 1 |
| 10-25 | 20 | 0 | 0 | 2 | 0 | 25 | 0 | 27 | 1 |
| 11- 1 | | | | | | | | | |
| 11- 8 | | | | | | | | | |
| 11-15 | | | | | | | | | |
| 11-22 | | | | | | | | | |
| 11-29 | | | | | | | | | |
| 12- 6 | | | | | | | | | |
| TOTAL | | 436 | 7733 | 387 | 4424 | 4777 | 0 | 17757 | 34 |

APPENDIX V. Sample run and result of report writer of CSDS (format 1)

```
$ RUN [SALMON.CATCH]CATCH
 ACCESS PROGRAM TO ONLINE CATCH STATISTICS
   1. CATCH BY AREA,PERIOD,GEAR,SPECIES
   2. CATCH BY YEAR,AREA,GEAR,SPECIES
   3. CATCH/BOAT WEEK BY AREA,PERIOD,SPECIES
   4. AVERAGE CATCH/BOAT/DAYS OPEN BY AREA, PERIOD,SPECIES
   WHICH (1-4)?

1
 ENTER STARTING YEAR (>=          51 ):

78
 ENTER ENDING YEAR (<=           80):

78

 1. CATCH BY PIECES
 2. CATCH BY WEIGHT
 WHICH (1 OR 2) ?

1

WANT TO SUM MORE THAN ONE AREA (Y/N)?

N

AREAS AVAILABLE:
1       2E      2W      3       4       5       6       7       8       9
10      30      11      12      13      14      15      16      17      18
19      20      21      22      23      24      25      26      27      28
C       29AB    29C     29D     29E     STIKINE TAKU    ALASKA  UNKNOWN 3X
3Y      3Z      5/1     5/2     5/3     5/4     5/6

ENTER # OF AREAS OR SUBAREAS WANTED  (1-44 OR ENTER 99 TO PRINT ALL AREAS):

1
ENTER THE  1 AREAS:
AREA #  1

4

PROCESSING.........

 TRY AGAIN (Y/N)?

N

 OUTPUT STORED IN FILE 'SCRAT.DAT'

 TYPE 'PRINT SCRAT.DAT' TO GET PRINTOUT

FORTRAN STOP
```

APPENDIX V (cont'd)

REVISED CATCH STATISTICS BY AREA IN PIECES 1978

AREA : 4

| | L. RED | | | M. RED | | | JACK | | | WHITE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WK. ENDING | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL |
| 3-11 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-18 | 0 | 0 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4- 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 4 |
| 4- 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 |
| 4-15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 |
| 4-22 | 0 | 0 | 3 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-29 | 0 | 0 | 15 | 0 | 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5- 6 | 0 | 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 2 |
| 5-13 | 0 | 0 | 9 | 0 | 0 | 18 | 0 | 0 | 16 | 0 | 0 | 5 |
| 5-20 | 0 | 0 | 45 | 0 | 0 | 29 | 0 | 0 | 22 | 1 | 0 | 10 |
| 5-27 | 0 | 0 | 87 | 0 | 0 | 29 | 0 | 0 | 7 | 1 | 0 | 21 |
| 6- 3 | 0 | 0 | 100 | 0 | 0 | 68 | 0 | 0 | 44 | 0 | 0 | 34 |
| 6-10 | 0 | 0 | 367 | 0 | 0 | 201 | 0 | 0 | 143 | 0 | 0 | 152 |
| 6-17 | 1 | 0 | 67 | 0 | 0 | 76 | 0 | 0 | 258 | 2 | 0 | 37 |
| 6-24 | 11 | 0 | 34 | 0 | 0 | 30 | 4 | 0 | 156 | 3 | 0 | 22 |
| 7- 1 | 37 | 0 | 220 | 0 | 0 | 224 | 6 | 0 | 194 | 5 | 0 | 102 |
| 7- 8 | 862 | 70 | 69 | 0 | 0 | 55 | 572 | 161 | 94 | 106 | 27 | 59 |
| 7-15 | 1191 | 292 | 13 | 0 | 0 | 1 | 1176 | 590 | 1 | 165 | 170 | 8 |
| 7-22 | 1023 | 11 | 11 | 0 | 0 | 7 | 529 | 35 | 36 | 104 | 7 | 2 |
| 7-29 | 663 | 101 | 28 | 0 | 0 | 10 | 519 | 247 | 33 | 48 | 55 | 2 |
| 8- 5 | 5 | 0 | 9 | 0 | 0 | 13 | 0 | 0 | 14 | 0 | 0 | 3 |
| 8-12 | 79 | 0 | 25 | 0 | 0 | 10 | 43 | 0 | 6 | 28 | 0 | 8 |
| 8-19 | 93 | 5 | 113 | 0 | 0 | 56 | 38 | 19 | 86 | 9 | 1 | 10 |
| 8-26 | 29 | 0 | 95 | 0 | 0 | 40 | 34 | 0 | 52 | 3 | 0 | 11 |
| 9- 2 | 19 | 0 | 127 | 0 | 0 | 126 | 31 | 0 | 187 | 10 | 0 | 16 |
| 9- 9 | 19 | 0 | 10 | 0 | 0 | 11 | 8 | 0 | 27 | 7 | 0 | 3 |
| 9-16 | 35 | 0 | 50 | 0 | 0 | 5 | 27 | 0 | 4 | 5 | 0 | 6 |
| 9-23 | 0 | 0 | 6 | 0 | 0 | 10 | 0 | 0 | 14 | 0 | 0 | 1 |
| 9-30 | 0 | 0 | 3 | 0 | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 1 |
| 10- 7 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| 10-14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11- 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12- 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D-F | 0 | 0 | 4 | 0 | 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 |
| TOTAL | 4067 | 479 | 1522 | 0 | 0 | 1034 | 2987 | 1052 | 1398 | 497 | 260 | 523 |

APPENDIX V (cont'd)

REVISED CATCH STATISTICS BY AREA IN PIECES 1978

AREA : 4

| | ALL SPRING | | | SOCKEYE | | | COHO | | | PINK | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WK. ENDING | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL |
| 3-11 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-18 | 0 | 0 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4- 1 | 0 | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4- 8 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-15 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-22 | 0 | 0 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-29 | 0 | 0 | 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5- 6 | 0 | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-13 | 0 | 0 | 48 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-20 | 1 | 0 | 106 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-27 | 1 | 0 | 144 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6- 3 | 0 | 0 | 246 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6-10 | 0 | 0 | 863 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6-17 | 3 | 0 | 438 | 0 | 0 | 0 | 0 | 0 | 37 | 0 | 0 | 36 |
| 6-24 | 18 | 0 | 242 | 187 | 0 | 0 | 0 | 0 | 128 | 0 | 0 | 2 |
| 7- 1 | 48 | 0 | 740 | 388 | 0 | 2 | 0 | 0 | 2923 | 0 | 0 | 15 |
| 7- 8 | 1540 | 258 | 277 | 49411 | 882 | 4 | 5976 | 419 | 1153 | 2362 | 5077 | 113 |
| 7-15 | 2532 | 1052 | 23 | 180907 | 6991 | 0 | 6638 | 2293 | 36 | 10369 | 24440 | 505 |
| 7-22 | 1656 | 53 | 56 | 89726 | 1641 | 2 | 3135 | 166 | 909 | 11910 | 7169 | 5 |
| 7-29 | 1230 | 403 | 73 | 55536 | 2957 | 24 | 7100 | 886 | 707 | 28940 | 29952 | 730 |
| 8- 5 | 5 | 0 | 39 | 214 | 0 | 7 | 10 | 0 | 581 | 501 | 0 | 4856 |
| 8-12 | 150 | 0 | 49 | 9180 | 0 | 7 | 7633 | 0 | 1170 | 51206 | 0 | 2123 |
| 8-19 | 140 | 25 | 265 | 6532 | 48 | 1 | 9877 | 77 | 1729 | 120163 | 9863 | 2102 |
| 8-26 | 66 | 0 | 198 | 1067 | 0 | 3 | 3356 | 0 | 1127 | 43879 | 0 | 1015 |
| 9- 2 | 60 | 0 | 456 | 184 | 0 | 0 | 2454 | 22 | 2869 | 11181 | 4504 | 1539 |
| 9- 9 | 34 | 0 | 51 | 69 | 0 | 0 | 5183 | 0 | 433 | 4188 | 0 | 589 |
| 9-16 | 67 | 0 | 65 | 27 | 0 | 0 | 3966 | 0 | 349 | 708 | 0 | 89 |
| 9-23 | 0 | 0 | 31 | 0 | 0 | 0 | 0 | 0 | 195 | 0 | 0 | 14 |
| 9-30 | 0 | 0 | 9 | 0 | 0 | 0 | 0 | 0 | 114 | 0 | 0 | 4 |
| 10- 7 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 13 | 0 | 0 | 3 |
| 10-14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11- 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12- 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D-F | 0 | 0 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TOTAL | 7551 | 1791 | 4477 | 393428 | 12519 | 51 | 55328 | 3863 | 14473 | 285407 | 81005 | 13742 |

APPENDIX V (cont'd)

REVISED CATCH STATISTICS BY AREA IN PIECES 1978

AREA : 4

| | CHUM | | | STEELHEAD | | | DAYS OPEN | | | DELIVERIES | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WK. ENDING | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL |
| 3-11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3-18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 |
| 3-25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4- 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4- 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 |
| 4-15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4-22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 14 |
| 4-29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10 |
| 5- 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| 5-13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 12 |
| 5-20 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 38 |
| 5-27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 32 |
| 6- 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 47 |
| 6-10 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 67 |
| 6-17 | 0 | 0 | 9 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 30 |
| 6-24 | 0 | 0 | 1 | 8 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 32 |
| 7- 1 | 0 | 0 | 1 | 5 | 0 | 0 | 0 | 0 | 0 | 5 | 0 | 63 |
| 7- 8 | 10721 | 1200 | 18 | 250 | 2 | 1 | 0 | 0 | 0 | 1017 | 8 | 57 |
| 7-15 | 14852 | 4504 | 3 | 521 | 9 | 0 | 0 | 0 | 0 | 1908 | 53 | 6 |
| 7-22 | 4782 | 278 | 1 | 391 | 1 | 0 | 0 | 0 | 0 | 695 | 6 | 20 |
| 7-29 | 9755 | 1291 | 41 | 594 | 11 | 0 | 0 | 0 | 0 | 1134 | 13 | 72 |
| 8- 5 | 21 | 0 | 60 | 30 | 0 | 1 | 0 | 0 | 0 | 5 | 0 | 40 |
| 8-12 | 2997 | 0 | 37 | 607 | 0 | 0 | 0 | 0 | 0 | 461 | 0 | 55 |
| 8-19 | 5561 | 53 | 41 | 1042 | 2 | 0 | 0 | 0 | 0 | 621 | 3 | 61 |
| 8-26 | 1383 | 0 | 52 | 265 | 0 | 2 | 0 | 0 | 0 | 302 | 0 | 98 |
| 9- 2 | 804 | 15 | 48 | 49 | 0 | 0 | 0 | 0 | 0 | 121 | 1 | 96 |
| 9- 9 | 1998 | 0 | 5 | 70 | 0 | 0 | 0 | 0 | 0 | 136 | 0 | 36 |
| 9-16 | 1247 | 0 | 15 | 38 | 0 | 1 | 0 | 0 | 0 | 199 | 0 | 27 |
| 9-23 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 33 |
| 9-30 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 18 |
| 10- 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 |
| 10-14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11- 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12- 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D-F | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 |
| TOTAL | 54121 | 7341 | 338 | 3870 | 25 | 5 | 0 | 0 | 0 | 6613 | 84 | 982 |

APPENDIX V - Format 2

```
$ RUN [SALMON.CATCH]CATCH
 ACCESS PROGRAM TO ONLINE CATCH STATISTICS
   1. CATCH BY AREA,PERIOD,GEAR,SPECIES
   2. CATCH BY YEAR,AREA,GEAR,SPECIES
   3. CATCH/BOAT WEEK BY AREA,PERIOD,SPECIES
   4. AVERAGE CATCH/BOAT/DAYS OPEN BY AREA, PERIOD,SPECIES
   WHICH (1-4)?

2
 ENTER STARTING YEAR (>=          51 ):

67

 ENTER ENDING YEAR (<=           80):

80

 1. CATCH BY PIECES
 2. CATCH BY WEIGHT
 WHICH (1 OR 2) ?

1

WANT TO SUM MORE THAN ONE AREA (Y/N)?

Y

 1. ANY COMBINATION OF AREAS OR SUBAREAS
 2. TOTAL B.C.
 3. NORTH COAST SUMMARY (STAT.AREA 1-10,30)
 4. SOUTH COAST SUMMARY (STAT.AREA 11-27,C)
 5. FRASER AREA SUMMARY (STAT.AREA 28+29)
 WHICH (1-5)?

1

AREAS AVAILABLE:
1       2E      2W      3       4       5       6       7       8       9
10      30      11      12      13      14      15      16      17      18
19      20      21      22      23      24      25      26      27      28
C       29AB    29C     29D     29E     STIKINE TAKU    ALASKA  UNKNOWN 3X
3Y      3Z      5/1     5/2     5/3     5/4     5/6

ENTER # OF AREAS OR SUBAREAS TO SUM (2-47):

8
ENTER THE  8 AREAS:
AREA #  1

13
AREA #  2
14
AREA #  3
15
```

APPENDIX V (cont'd)

```
AREA #  4
16
AREA #  5
17
AREA #  6
18
AREA #  7
19
AREA #  8
28
ENTER NAME OF AREA COMBINATION (E.G. GEORGIA STRAIT)
GEORGIA STRAIT

PROCESSING.........

 TRY AGAIN (Y/N)?
N

 OUTPUT STORED IN FILE 'SCRAT.DAT'

 TYPE 'PRINT SCRAT.DAT' TO GET PRINTOUT

FORTRAN STOP
```

APPENDIX V (cont'd)

CATCH BY SPECIS AND GEAR IN PIECES 1967-1980

AREA : GEORGIA STRAIT

| | L. RED | | | M. RED | | | JACK | | | WHITE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YEAR | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL |
| 1980 | 1420 | 8310 | 12690 | 0 | 0 | 27330 | 750 | 6670 | 218950 | 90 | 180 | 13640 |
| 1979 | 1690 | 6660 | 11890 | 0 | 0 | 23850 | 170 | 4470 | 210730 | 860 | 90 | 6950 |
| 1978 | 1670 | 8030 | 14100 | 0 | 0 | 25600 | 2930 | 9130 | 162020 | 710 | 460 | 7330 |
| 1977 | 1430 | 11150 | 6540 | 0 | 0 | 15770 | 330 | 9190 | 196730 | 4650 | 680 | 22500 |
| 1976 | 1700 | 4820 | 9150 | 0 | 0 | 22470 | 530 | 9780 | 148590 | 2140 | 750 | 13400 |
| 1975 | 1650 | 6480 | 6250 | 0 | 0 | 13500 | 710 | 5620 | 128430 | 440 | 700 | 9280 |
| 1974 | 1210 | 5540 | 9470 | 0 | 0 | 15430 | 480 | 4040 | 111530 | 980 | 130 | 12710 |
| 1973 | 3160 | 4920 | 8040 | 0 | 0 | 13830 | 840 | 4970 | 103260 | 670 | 120 | 15830 |
| 1972 | 1960 | 3510 | 15010 | 0 | 0 | 20530 | 410 | 2970 | 167080 | 670 | 120 | 17650 |
| 1971 | 2730 | 4550 | 9950 | 0 | 0 | 22530 | 840 | 3670 | 226710 | 850 | 450 | 16690 |
| 1970 | 4580 | 3170 | 7660 | 0 | 0 | 12870 | 2050 | 3470 | 84080 | 2150 | 350 | 15260 |
| 1969 | 2000 | 3360 | 7000 | 0 | 0 | 15090 | 280 | 2170 | 65300 | 970 | 830 | 9200 |
| 1968 | 3040 | 1140 | 6940 | 0 | 0 | 12490 | 310 | 510 | 53230 | 1160 | 210 | 10140 |
| 1967 | 3240 | 3240 | 7960 | 0 | 0 | 20600 | 220 | 1040 | 82450 | 1300 | 650 | 11070 |

APPENDIX V (cont'd)

CATCH BY SPECIS AND GEAR IN PIECES 1967-1980

AREA : GEORGIA STRAIT

| | ALL SPRING | | | SOCKEYE | | | COHO | | | PINK | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YEAR | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL |
| 1980 | 2260 | 15160 | 272610 | 19070 | 380230 | 6170 | 8940 | 39360 | 150690 | 1070 | 74880 | 3280 |
| 1979 | 2720 | 11220 | 253420 | 27500 | 489190 | 24750 | 40430 | 22520 | 222560 | 6830 | 308000 | 34290 |
| 1978 | 5310 | 17620 | 209050 | 30880 | 883110 | 63120 | 42890 | 41540 | 321510 | 3730 | 89560 | 6030 |
| 1977 | 6410 | 21020 | 241540 | 47700 | 279860 | 9150 | 30690 | 51530 | 140670 | 11870 | 422700 | 24430 |
| 1976 | 4370 | 15350 | 193610 | 23620 | 182360 | 19090 | 31200 | 25380 | 72510 | 1120 | 122130 | 1900 |
| 1975 | 2800 | 12800 | 157460 | 6730 | 30800 | 23860 | 15260 | 36930 | 111450 | 20630 | 392700 | 22450 |
| 1974 | 2670 | 9710 | 149140 | 64670 | 416820 | 63580 | 15890 | 20550 | 147580 | 1720 | 172170 | 2680 |
| 1973 | 4670 | 10010 | 140960 | 39440 | 69960 | 11590 | 27050 | 28170 | 92340 | 24390 | 416540 | 19780 |
| 1972 | 3040 | 6600 | 220270 | 48760 | 78490 | 5600 | 15340 | 11640 | 62680 | 670 | 87280 | 1130 |
| 1971 | 4420 | 8670 | 275880 | 45600 | 219250 | 25620 | 19270 | 59460 | 238160 | 48830 | 900380 | 27090 |
| 1970 | 8780 | 6990 | 119870 | 37750 | 141050 | 3950 | 30130 | 27430 | 161630 | 24650 | 160250 | 1450 |
| 1969 | 3250 | 6360 | 96590 | 35550 | 97570 | 3220 | 4880 | 8900 | 41290 | 13650 | 140000 | 12230 |
| 1968 | 4510 | 1860 | 82800 | 27460 | 27010 | 1240 | 10730 | 12570 | 118170 | 3960 | 214380 | 15160 |
| 1967 | 4760 | 4930 | 122080 | 53230 | 226680 | 7590 | 15130 | 27700 | 142660 | 36260 | 669300 | 27270 |

APPENDIX V (cont'd)

CATCH BY SPECIS AND GEAR IN PIECES 1967-1980

AREA : GEORGIA STRAIT

| | CHUM | | | STEELHEAD | | | DAYS OPEN | | | DELIVERIES | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| YEAR | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL | GN | SEINE | TROLL |
| 1980 | 71910 | 327680 | 1090 | 0 | 40 | 0 | 0 | 0 | 0 | 1510 | 2040 | 27000 |
| 1979 | 6340 | 21020 | 660 | 30 | 20 | 30 | 0 | 0 | 0 | 910 | 1640 | 26280 |
| 1978 | 103580 | 527210 | 5820 | 60 | 80 | 60 | 0 | 0 | 0 | 2780 | 2740 | 26140 |
| 1977 | 5610 | 35170 | 1250 | 30 | 60 | 0 | 0 | 0 | 0 | 1790 | 1850 | 25590 |
| 1976 | 99160 | 326160 | 2830 | 10 | 780 | 20 | 0 | 0 | 0 | 3910 | 2050 | 18980 |
| 1975 | 62500 | 136360 | 1550 | 30 | 0 | 20 | 0 | 0 | 0 | 2360 | 1870 | 15300 |
| 1974 | 44150 | 97000 | 3140 | 20 | 0 | 30 | 0 | 0 | 0 | 2770 | 1590 | 16410 |
| 1973 | 415640 | 1144740 | 530 | 80 | 10 | 0 | 0 | 0 | 0 | 8620 | 3260 | 16450 |
| 1972 | 291910 | 642670 | 90 | 70 | 20 | 0 | 0 | 0 | 0 | 5880 | 2050 | 18760 |
| 1971 | 5700 | 15180 | 100 | 40 | 20 | 20 | 0 | 0 | 0 | 2330 | 1840 | 26820 |
| 1970 | 162400 | 274840 | 410 | 40 | 0 | 0 | 0 | 0 | 0 | 6530 | 1840 | 20180 |
| 1969 | 110530 | 193350 | 110 | 60 | 10 | 0 | 0 | 0 | 0 | 3840 | 1730 | 15050 |
| 1968 | 77940 | 201970 | 60 | 120 | 0 | 20 | 0 | 0 | 0 | 2550 | 820 | 17730 |
| 1967 | 11670 | 49940 | 190 | 60 | 10 | 20 | 0 | 0 | 0 | 2280 | 1480 | 21580 |

APPENDIX V - Format 3

```
$ RUN [SALMON.CATCH]CATCH
 ACCESS PROGRAM TO ONLINE CATCH STATISTICS
   1. CATCH BY AREA,PERIOD,GEAR,SPECIES
   2. CATCH BY YEAR,AREA,GEAR,SPECIES
   3. CATCH/BOAT WEEK BY AREA,PERIOD,SPECIES
   4. AVERAGE CATCH/BOAT/DAYS OPEN BY AREA, PERIOD,SPECIES
   WHICH (1-4)?
3
 ENTER STARTING YEAR (>=          67 ):
75
 ENTER ENDING YEAR (<=          80):
80

 1. CATCH BY PIECES
 2. CATCH BY WEIGHT
 WHICH (1 OR 2) ?
2

WANT TO SUM MORE THAN ONE AREA (Y/N)?
N

AREAS AVAILABLE:
1       2E      2W      3       4       5       6       7       8       9
10      30      11      12      13      14      15      16      17      18
19      20      21      22      23      24      25      26      27      28
C       29AB    29C     29D     29E     STIKINE TAKU    ALASKA  UNKNOWN 3X
3Y      3Z      5/1     5/2     5/3     5/4     5/6

ENTER # OF AREAS OR SUBAREAS WANTED  (1-44 OR ENTER 99 TO PRINT ALL AREAS):
1
ENTER THE  1 AREAS:
AREA #  1
4
```

APPENDIX V (cont'd)

```
  1. GILL NET
  2. SEINE
  3. TROLL
  4. TROLL + FR. TROLL
  WHICH (1-4)

1

 1.   L. RED
 2.   M. RED
 3.    JACK
 4.   WHITE
 5. ALL SPRING
 6.   SOCKEYE
 7.    COHO
 8.    PINK
 9.    CHUM
10.  STEELHEAD
 11. SPECIES 5 TO 9 INCLUSIVE
 WHICH (1-11)?

6

PROCESSING.....

 TRY AGAIN (Y/N)?

N

 OUTPUT STORED IN FILE 'SCRAT.DAT'

 TYPE 'PRINT SCRAT.DAT' TO GET PRINTOUT

FORTRAN STOP
$
```

APPENDIX V (cont'd)

SOCKEYE CATCH/ GN BOAT WEEK IN WEIGHT 1975-1980

AREA : 4

| PERIOD | 80 | 79 | 78 | 77 | 76 | 75 |
|---|---|---|---|---|---|---|
| 3-1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3-4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4-4 | 0 | 0 | 0 | 374 | 0 | 0 |
| 4-5 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5-4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6-1 | 0 | 0 | 0 | 0 | 1661 | 0 |
| 6-2 | 0 | 208 | 0 | 0 | 7824 | 0 |
| 6-3 | 0 | 968 | 1128 | 0 | 83 | 410 |
| 6-4 | 0 | 7361 | 2333 | 326 | 210 | 352 |
| 7-1 | 110 | 11673 | 355139 | 3299 | 2345 | 288414 |
| 7-2 | 638647 | 477013 | 1295148 | 264316 | 113100 | 201188 |
| 7-3 | 721347 | 1514772 | 625259 | 1209370 | 174887 | 1712 |
| 7-4 | 252626 | 2021168 | 367339 | 849511 | 736622 | 850016 |
| 7-5 | 491718 | 839837 | 1338 | 1776113 | 1074205 | 473430 |
| 8-1 | 3326 | 1498815 | 57479 | 892005 | 1019654 | 390901 |
| 8-2 | 0 | 224182 | 40254 | 118502 | 111870 | 199211 |
| 8-3 | 300 | 74 | 6437 | 23050 | 11064 | 14369 |
| 8-4 | 50 | 2 | 1098 | 5855 | 31624 | 4542 |
| 9-1 | 942 | 0 | 466 | 70 | 76783 | 340 |
| 9-2 | 0 | 0 | 193 | 61 | 18402 | 0 |
| 9-3 | 0 | 0 | 0 | 0 | 15866 | 0 |
| 9-4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10-5 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D-F | 0 | 0 | 0 | 0 | 0 | 0 |

APPENDIX VI. Example of interfacing user program with CSDS Subroutines.

```
C......PROGRAM TO REDISTRIBUTE SPRING AND COHO FREEZER BOAT LANDINGS
C         ACCORDING TO DAY BOAT LANDINGS
C
      IMPLICIT INTEGER*4(C)
      COMMON  C(11,5),CA(10,41,3)
      INTEGER*4 TDF(5),TGR(5),DO(3)
      CHARACTER YR*2,AR*7
  902 FORMAT(A2)
  903 FORMAT(I2)
  904 FORMAT(A7)
      OPEN(10,FILE='SCRAT.DAT',STATUS='NEW')
   11 PRINT *,' ENTER YEAR : '
      READ(5,903,ERR=11)IYR
      ENCODE(2,903,YR)IYR
   12 PRINT *,'ENTER AREA : '
      READ(5,904,ERR=12)AR
      CALL GLIB1(AR,IAR,0)
      CALL GFO(IYR,1)
C....LOOP OVER PERIOD
      DO 1 IPR=1,40
      CALL CAA1(IAR,IPR,1,C,TGR,TDF,DO)
      CALL OM(IPR)
    1 CONTINUE
C....REDISTRIBUTE FREEZER BOAT CATCHES
      DO 21 IP=1,3
      DO 22 I=1,2
      IO=I*5
      J=IO-2
      J1=IO-1
      IF(CA(J,41,IP).EQ.0)GOTO 22
      DO 23 IPR=1,41
      CA(IO,IPR,IP)=FLOAT(CA(J,IPR,IP))/FLOAT(CA(J,41,IP))
     . *CA(J1,41,IP)
   23 CONTINUE
   22 CONTINUE
   21 CONTINUE
C....PRINTOUT
      CALL OF(YR,AR)
      STOP
      END



C
```

APPENDIX VI (cont'd)

```
C.....SUBROUTINE TO FORM OUTPUT MATRIX
      SUBROUTINE OM(IPR)
      IMPLICIT INTEGER*4(C)
      COMMON  C(11,5),CA(10,41,3)
      DO 1 IGR=1,4
C...FIRST PAGE
      DO 2 ISP=1,2
      IO=(ISP-1)*5+IGR
      CA(IO,IPR,1)=C(ISP,IGR)
      CA(IO,41,1)=CA(IO,41,1)+C(ISP,IGR)
C...ALL SPRINGS
      IF(IO.GT.5)IO=IO-5
      CA(IO,IPR,3)=CA(IO,IPR,3)+C(ISP,IGR)
      CA(IO,41,3)=CA(IO,41,3)+C(ISP,IGR)
   2  CONTINUE
C
C....SECOND PAGE
      DO 3 ISP=3,5
      IS=ISP
      IF(IS.GE.4)IS=IS-1
      IO=(IS-3)*5+IGR
      CA(IO,IPR,2)=CA(IO,IPR,2)+C(ISP,IGR)
      CA(IO,41,2)=CA(IO,41,2)+C(ISP,IGR)
C... ALL SPRINGS
      IF(IO.GT.5)IO=IO-5
      CA(IO,IPR,3)=CA(IO,IPR,3)+C(ISP,IGR)
      CA(IO,41,3)=CA(IO,41,3)+C(ISP,IGR)
  3   CONTINUE
C
C....THIRD PAGE
      ISP=7
      IO=IGR+5
      CA(IO,IPR,3)=C(ISP,IGR)
      CA(IO,41,3)=CA(IO,41,3)+C(ISP,IGR)
  1   CONTINUE
      RETURN
      END



C
C
C.....SUBROUTINE TO PRINT OUTPUT
      SUBROUTINE OF(YR,AR)
      IMPLICIT INTEGER*4(C)
```

APPENDIX VI (cont'd)

```
      COMMON  C(11,5),CA(10,41,3)
      CHARACTER YR*2,AR*7
      CHARACTER*10 SP(6),GR(5)
      DATA SP/'  L.RED','  M.RED','   JACK','  WHITE','ALL SPRING',
     .   '   COHO'/
      DATA GR/'    GN','  SEINE','  TROLL',' FR.TR.',' FR.TR.2'/
  900 FORMAT('1',40X,' CATCH STATISTICS BY AREA IN PIECES  19',A2)
  901 FORMAT('   AREA : ',A7)
  902 FORMAT(8X,2(15X,A10,15X))
  903 FORMAT(' WK. ENDING ',10A8)
  904 FORMAT(1X,I2,'-',I2,2X,10I8)
  905 FORMAT('  TOTAL  ',10I8)
  906 FORMAT(' ** FR.TR.2 = REDISTRIBUTED FREEZER TROLL CATCHES')
C.....LOOP OVER PAGES
      DO 1 IP=1,3
      WRITE(10,900)YR
      WRITE(10,*)
      WRITE(10,901)AR
      WRITE(10,*)
      J=(IP-1)*2+1
      WRITE(10,902) (SP(I),I=J,J+1)
      WRITE(10,*)
      WRITE(10,903) ((GR(I),I=1,5),J=1,2)
      WRITE(10,*)
      DO 2 IPR=1,40
      CALL GLIB2(YR,IPR,MO,IDAY)
    2 WRITE(10,904)MO,IDAY,(CA(J,IPR,IP),J=1,10)
      WRITE(10,*)
      WRITE(10,905)(CA(J,41,IP),J=1,10)
      WRITE(10,*)
      WRITE(10,906)
    1 CONTINUE
      RETURN
      END
```

APPENDIX VI (cont'd)

```
$FORTRAN FREEZER
$LINK FREEZER+[SALMON.CATCH]CATCH/LIB
$RUN FREEZER
 ENTER YEAR:
 78
 ENTER AREA:
 23
$FORTRAN STOP
PRINT SCRAT.DAT
```

APPENDIX VI (cont'd)

CATCH STATISTICS BY AREA IN PIECES 1978

AREA : 23

| | JACK | | | | | WHITE | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| WK. ENDING | GN | SEINE | TROLL | FR.TR. | FR.TR.2 | GN | SEINE | TROLL | FR.TR. | FR.TR.2 |
| 3-11 | 0 | 0 | 771 | 0 | 47 | 0 | 0 | 3 | 0 | 1 |
| 3-18 | 0 | 0 | 879 | 0 | 53 | 0 | 0 | 5 | 0 | 2 |
| 3-25 | 0 | 0 | 561 | 0 | 34 | 0 | 0 | 12 | 0 | 5 |
| 4- 1 | 0 | 0 | 608 | 0 | 37 | 0 | 0 | 4 | 0 | 1 |
| 4- 8 | 0 | 0 | 193 | 0 | 11 | 0 | 0 | 2 | 0 | 0 |
| 4-15 | 0 | 0 | 218 | 0 | 13 | 0 | 0 | 0 | 0 | 0 |
| 4-22 | 0 | 0 | 3632 | 0 | 222 | 0 | 0 | 54 | 0 | 24 |
| 4-29 | 0 | 0 | 5519 | 0 | 338 | 0 | 0 | 419 | 0 | 188 |
| 5- 6 | 0 | 0 | 3815 | 90 | 233 | 0 | 0 | 54 | 11 | 24 |
| 5-13 | 0 | 0 | 4638 | 108 | 284 | 0 | 0 | 842 | 41 | 379 |
| 5-20 | 0 | 0 | 4444 | 0 | 272 | 11 | 0 | 189 | 0 | 85 |
| 5-27 | 0 | 0 | 4775 | 28 | 292 | 55 | 0 | 437 | 11 | 196 |
| 6- 3 | 10 | 81 | 4782 | 159 | 293 | 35 | 0 | 504 | 396 | 227 |
| 6-10 | 358 | 1827 | 5742 | 10 | 351 | 10 | 5 | 1868 | 92 | 842 |
| 6-17 | 340 | 2821 | 2953 | 83 | 180 | 12 | 6 | 273 | 224 | 123 |
| 6-24 | 806 | 4304 | 5227 | 0 | 320 | 14 | 48 | 471 | 0 | 212 |
| 7- 1 | 519 | 4392 | 6281 | 175 | 384 | 2 | 67 | 988 | 173 | 445 |
| 7- 8 | 511 | 780 | 5065 | 137 | 310 | 2 | 2 | 279 | 411 | 125 |
| 7-15 | 155 | 196 | 6743 | 459 | 413 | 2 | 0 | 898 | 803 | 404 |
| 7-22 | 97 | 178 | 1663 | 532 | 101 | 0 | 1 | 161 | 384 | 72 |
| 7-29 | 0 | 0 | 3103 | 369 | 190 | 2 | 0 | 944 | 568 | 425 |
| 8- 5 | 0 | 0 | 1321 | 0 | 80 | 0 | 0 | 335 | 0 | 151 |
| 8-12 | 0 | 0 | 1361 | 86 | 83 | 0 | 0 | 184 | 194 | 82 |
| 8-19 | 0 | 0 | 891 | 77 | 54 | 0 | 0 | 265 | 112 | 119 |
| 8-26 | 0 | 0 | 1332 | 32 | 81 | 3 | 0 | 394 | 148 | 177 |
| 9- 2 | 76 | 0 | 1851 | 0 | 113 | 0 | 0 | 93 | 0 | 41 |
| 9- 9 | 446 | 0 | 3573 | 142 | 218 | 56 | 0 | 280 | 52 | 126 |
| 9-16 | 348 | 0 | 1114 | 168 | 68 | 4 | 0 | 59 | 137 | 26 |
| 9-23 | 42 | 0 | 1670 | 391 | 102 | 0 | 25 | 157 | 277 | 70 |
| 9-30 | 32 | 0 | 981 | 410 | 60 | 0 | 0 | 105 | 126 | 47 |
| 10- 7 | 307 | 0 | 602 | 351 | 36 | 0 | 0 | 50 | 241 | 22 |
| 10-14 | 506 | 0 | 465 | 380 | 28 | 0 | 0 | 23 | 72 | 10 |
| 10-21 | 1302 | 11 | 1289 | 189 | 78 | 0 | 0 | 157 | 50 | 70 |
| 10-28 | 2320 | 23 | 249 | 466 | 15 | 0 | 0 | 40 | 233 | 18 |
| 11- 4 | 1025 | 0 | 950 | 772 | 58 | 0 | 0 | 154 | 71 | 69 |
| 11-11 | 513 | 0 | 19 | 0 | 1 | 0 | 0 | 3 | 0 | 1 |
| 11-18 | 0 | 0 | 26 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 11-25 | 0 | 0 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12- 2 | 0 | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12- 9 | 0 | 0 | 2294 | 0 | 140 | 0 | 0 | 2 | 0 | 0 |
| TOTAL | 9713 | 14613 | 91616 | 5614 | 5614 | 208 | 154 | 10708 | 4827 | 4827 |

** FR.TR.2 = REDISTRIBUTED FREEZER TROLL CATCHES

APPENDIX VI (cont'd)

CATCH STATISTICS BY AREA IN PIECES 1978

AREA : 23

| | L.RED | | | | | M.RED | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| WK. ENDING | GN | SEINE | TROLL | FR.TR. | FR.TR.2 | GN | SEINE | TROLL | FR.TR. | FR.TR.2 |
| 3-11 | 0 | 0 | 172 | 0 | 17 | 0 | 0 | 341 | 0 | 33 |
| 3-18 | 0 | 0 | 126 | 0 | 12 | 0 | 0 | 443 | 0 | 43 |
| 3-25 | 0 | 0 | 139 | 0 | 13 | 0 | 0 | 343 | 0 | 33 |
| 4- 1 | 0 | 0 | 118 | 0 | 11 | 0 | 0 | 263 | 0 | 25 |
| 4- 8 | 0 | 0 | 48 | 0 | 4 | 0 | 0 | 64 | 0 | 6 |
| 4-15 | 0 | 0 | 61 | 0 | 6 | 0 | 0 | 80 | 0 | 7 |
| 4-22 | 0 | 0 | 985 | 0 | 98 | 0 | 0 | 2977 | 0 | 292 |
| 4-29 | 0 | 0 | 2750 | 0 | 275 | 0 | 0 | 5505 | 0 | 541 |
| 5- 6 | 0 | 0 | 1668 | 69 | 167 | 0 | 0 | 2387 | 104 | 234 |
| 5-13 | 0 | 0 | 5364 | 177 | 538 | 0 | 0 | 6108 | 227 | 600 |
| 5-20 | 8 | 0 | 3569 | 0 | 358 | 0 | 0 | 4470 | 0 | 439 |
| 5-27 | 38 | 0 | 5000 | 49 | 501 | 0 | 0 | 5193 | 81 | 510 |
| 6- 3 | 92 | 14 | 4595 | 339 | 461 | 0 | 0 | 4623 | 425 | 454 |
| 6-10 | 225 | 668 | 7179 | 344 | 720 | 0 | 0 | 6124 | 181 | 602 |
| 6-17 | 303 | 631 | 2739 | 224 | 274 | 0 | 0 | 2478 | 119 | 243 |
| 6-24 | 605 | 510 | 5952 | 0 | 597 | 0 | 0 | 4497 | 0 | 442 |
| 7- 1 | 262 | 388 | 7253 | 263 | 727 | 0 | 0 | 4134 | 267 | 406 |
| 7- 8 | 301 | 291 | 7486 | 399 | 751 | 0 | 0 | 3448 | 345 | 339 |
| 7-15 | 88 | 157 | 10418 | 2305 | 1045 | 0 | 0 | 4837 | 977 | 475 |
| 7-22 | 37 | 18 | 2227 | 1005 | 223 | 0 | 0 | 942 | 670 | 92 |
| 7-29 | 0 | 0 | 9455 | 937 | 948 | 0 | 0 | 4010 | 509 | 394 |
| 8- 5 | 0 | 0 | 3297 | 0 | 330 | 0 | 0 | 1069 | 0 | 105 |
| 8-12 | 0 | 0 | 3670 | 371 | 368 | 0 | 0 | 1535 | 174 | 150 |
| 8-19 | 0 | 0 | 2752 | 266 | 276 | 0 | 0 | 1147 | 99 | 112 |
| 8-26 | 0 | 0 | 2085 | 154 | 209 | 0 | 0 | 1199 | 59 | 117 |
| 9- 2 | 496 | 0 | 1831 | 0 | 183 | 0 | 0 | 1038 | 0 | 102 |
| 9- 9 | 4626 | 0 | 1770 | 71 | 177 | 0 | 0 | 1830 | 1 | 179 |
| 9-16 | 3526 | 0 | 694 | 476 | 69 | 0 | 0 | 708 | 247 | 69 |
| 9-23 | 30 | 0 | 659 | 499 | 66 | 0 | 0 | 1097 | 514 | 107 |
| 9-30 | 99 | 0 | 438 | 476 | 43 | 0 | 0 | 836 | 307 | 82 |
| 10- 7 | 421 | 0 | 287 | 186 | 28 | 0 | 0 | 494 | 261 | 48 |
| 10-14 | 124 | 0 | 135 | 145 | 13 | 0 | 0 | 387 | 346 | 38 |
| 10-21 | 2205 | 0 | 558 | 148 | 55 | 0 | 0 | 1308 | 225 | 128 |
| 10-28 | 4585 | 1 | 118 | 388 | 11 | 0 | 0 | 286 | 743 | 28 |
| 11- 4 | 2746 | 0 | 306 | 370 | 30 | 0 | 0 | 884 | 723 | 86 |
| 11-11 | 1012 | 0 | 5 | 0 | 0 | 0 | 0 | 34 | 0 | 3 |
| 11-18 | 0 | 0 | 12 | 0 | 1 | 0 | 0 | 40 | 0 | 3 |
| 11-25 | 0 | 0 | 7 | 0 | 0 | 0 | 0 | 15 | 78 | 1 |
| 12- 2 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 26 | 0 | 2 |
| 12- 9 | 0 | 0 | 339 | 0 | 34 | 0 | 0 | 918 | 0 | 90 |
| TOTAL | 21829 | 2678 | 96275 | 9661 | 9661 | 0 | 0 | 78118 | 7682 | 7682 |

** FR.TR.2 = REDISTRIBUTED FREEZER TROLL CATCHES

APPENDIX VI (cont'd)

CATCH STATISTICS BY AREA IN PIECES 1978

AREA : 23

| | ALL SPRING | | | | | COHO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| WK. ENDING | GN | SEINE | TROLL | FR.TR. | FR.TR.2 | GN | SEINE | TROLL | FR.TR. | FR.TR.2 |
| 3-11 | 0 | 0 | 1287 | 0 | 129 | 0 | 0 | 0 | 0 | 0 |
| 3-18 | 0 | 0 | 1453 | 0 | 145 | 0 | 0 | 0 | 0 | 0 |
| 3-25 | 0 | 0 | 1055 | 0 | 105 | 0 | 0 | 0 | 0 | 0 |
| 4- 1 | 0 | 0 | 993 | 0 | 99 | 0 | 0 | 0 | 0 | 0 |
| 4- 8 | 0 | 0 | 307 | 0 | 30 | 0 | 0 | 0 | 0 | 0 |
| 4-15 | 0 | 0 | 359 | 0 | 36 | 0 | 0 | 0 | 0 | 0 |
| 4-22 | 0 | 0 | 7648 | 0 | 767 | 0 | 0 | 0 | 0 | 0 |
| 4-29 | 0 | 0 | 14193 | 0 | 1425 | 0 | 0 | 0 | 0 | 0 |
| 5- 6 | 0 | 0 | 7924 | 274 | 795 | 0 | 0 | 0 | 0 | 0 |
| 5-13 | 0 | 0 | 16952 | 553 | 1702 | 0 | 0 | 0 | 0 | 0 |
| 5-20 | 19 | 0 | 12672 | 0 | 1272 | 0 | 0 | 0 | 0 | 0 |
| 5-27 | 93 | 0 | 15405 | 169 | 1546 | 0 | 0 | 0 | 0 | 0 |
| 6- 3 | 137 | 95 | 14504 | 1319 | 1456 | 8 | 2 | 0 | 0 | 0 |
| 6-10 | 593 | 2500 | 20913 | 627 | 2099 | 13 | 11 | 73 | 0 | 13 |
| 6-17 | 655 | 3458 | 8443 | 650 | 847 | 106 | 26 | 10822 | 0 | 2049 |
| 6-24 | 1425 | 4862 | 16147 | 0 | 1621 | 418 | 53 | 77793 | 0 | 14734 |
| 7- 1 | 783 | 4847 | 18656 | 878 | 1873 | 129 | 38 | 64677 | 4061 | 12250 |
| 7- 8 | 814 | 1073 | 16278 | 1292 | 1634 | 316 | 12 | 50890 | 4043 | 9638 |
| 7-15 | 245 | 353 | 22896 | 4544 | 2298 | 68 | 3 | 44111 | 18716 | 8354 |
| 7-22 | 134 | 197 | 4993 | 2591 | 501 | 14 | 1 | 8486 | 15507 | 1607 |
| 7-29 | 2 | 0 | 17512 | 2383 | 1758 | 3 | 0 | 16227 | 2793 | 3073 |
| 8- 5 | 0 | 0 | 6022 | 0 | 604 | 0 | 0 | 5236 | 0 | 991 |
| 8-12 | 0 | 0 | 6750 | 825 | 677 | 0 | 0 | 2277 | 2388 | 431 |
| 8-19 | 0 | 0 | 5055 | 554 | 507 | 0 | 0 | 2137 | 996 | 404 |
| 8-26 | 3 | 0 | 5010 | 393 | 503 | 0 | 0 | 3030 | 1228 | 573 |
| 9- 2 | 572 | 0 | 4813 | 0 | 483 | 265 | 0 | 19322 | 0 | 3659 |
| 9- 9 | 5128 | 0 | 7453 | 266 | 748 | 1712 | 179 | 62636 | 819 | 11863 |
| 9-16 | 3878 | 0 | 2575 | 1028 | 258 | 7823 | 0 | 18951 | 2985 | 3589 |
| 9-23 | 72 | 25 | 3583 | 1681 | 359 | 4640 | 336 | 19999 | 8662 | 3787 |
| 9-30 | 131 | 0 | 2360 | 1319 | 236 | 4927 | 0 | 4344 | 3689 | 822 |
| 10- 7 | 728 | 0 | 1433 | 1039 | 143 | 1956 | 0 | 2752 | 3665 | 521 |
| 10-14 | 630 | 0 | 1010 | 943 | 101 | 840 | 0 | 689 | 2460 | 130 |
| 10-21 | 3507 | 11 | 3312 | 612 | 332 | 260 | 80 | 955 | 1635 | 180 |
| 10-28 | 6905 | 24 | 693 | 1830 | 69 | 248 | 106 | 75 | 2965 | 14 |
| 11- 4 | 3771 | 0 | 2294 | 1936 | 230 | 36 | 1 | 153 | 2112 | 28 |
| 11-11 | 1525 | 0 | 61 | 0 | 6 | 333 | 0 | 0 | 0 | 0 |
| 11-18 | 0 | 0 | 78 | 0 | 7 | 0 | 0 | 0 | 0 | 0 |
| 11-25 | 0 | 0 | 33 | 78 | 3 | 0 | 0 | 0 | 0 | 0 |
| 12- 2 | 0 | 0 | 39 | 0 | 3 | 0 | 0 | 0 | 0 | 0 |
| 12- 9 | 0 | 0 | 3553 | 0 | 356 | 0 | 0 | 0 | 0 | 0 |
| TOTAL | 31750 | 17445 | 276717 | 27784 | 27784 | 24115 | 848 | 415635 | 78724 | 78724 |

## FR.TR.2 = REDISTRIBUTED FREEZER TROLL CATCHES

Listing A - Listing of program for translating original sales-slip records (SALESLIP.BAS)

```
100 PRINT' TO COPY SALESLIP TAPE FROM VANCOUVER TO VAX-11 TAPE: ' \PRINT
110 PRINT' -- FIRST DUMP CONTENT OF TAPE TO A FILE'
120 PRINT' -- THEN RUN THIS PROGRAM TO SCRAMBLE CFV AND CHANGE'
130 PRINT'      COBOL CHARACTERS TO #S (I.E. @-I TO 0-9) IN COL.20,40,48,56,72'
140 PRINT' -- COPY TRANSLATED VERSION TO VAXTAPE'\ PRINT
150 PRINT'          BY FRED WONG, NOV.1981'
155 PRINT\INPUT 'ARE YOU READY TO RUN THIS PROGRAM ?(Y/N)';DU$\ PRINT
160 IF DU$='Y' THEN GOTO 180
170 STOP
180 INPUT 'ENTER FILENAME OF ORIGINAL DATA: ';F1$
190 INPUT 'ENTER FILENAME FOR OUTPUT: ';F2$
195 INPUT 'ENTER ESTIMATED BLOCKS NEEDED FOR OUTPUT:'; C%
200 INPUT 'ENTER FILENAME FOR ERROR MESSAGES: ';F3$
210 OPEN F1$ FOR INPUT AS FILE #1
220 OPEN F2$ FOR OUTPUT AS FILE #2, FILESIZE C%, ORGANIZATION SEQUENTIAL FIXED, RECORDSIZE 72%
230 OPEN F3$ FOR OUTPUT AS FILE #3
235 CO%(1)=20\CO%(2)= 40\CO%(3)=48\CO%(4)=56\CO%(5)=72
240 ON ERROR GOTO 1000
250 INPUT #1,A$
251 J%=LEN(A$)
252 IF J%=72 GOTO 260
253 FOR I=J%+1 TO 72
254 A$=A$+'0'
255 NEXT I
260 IN%=IN%+1
270 GOSUB 400\ REM TRANSLATION ROUTINE
280 GOSUB 500 \REM SCRAMBLE CFV
290 DU$=MID$(A$,1,19)+A1$(1)+MID$(A$,21,6)+CFV$+MID$(A$,32,8)+A1$(2)
295 DU$=DU$+MID$(A$,41,7)+A1$(3)+MID$(A$,49,7)+A1$(4)+MID$(A$,57,15)+A1$(5)
300 PRINT #2,DU$
320 GOTO 250
400 REM TRANSLATION
420 FOR I=1 TO 5
430 J%=ASCII(MID$(A$,CO%(I),1))
440 IF J%>63 AND J%<74 THEN J%=J%-16
450 A1$(I)=CHR$(J%)
460 NEXT I
470 RETURN
500 REM SCRAMBLE CFV
510 ER=0 \REM ERROR FLAG
520 CFV$=MID$(A$,27,5)
530 CALL SCRAMBLE(CFV$, ER)
540 IF ER=1 THEN PRINT #3,'IN RECORD: ';IN%
550 RETURN
1000 RESUME 1200
```

Listing A (cont'd)

```
1200 PRINT 'TOTAL # OF RECORDS CONVERTED: ';INX
1250 END
```

Listing B - Listing of program for generating
CDDS files (DB.FOR)

```
C.....PROGRAM TO SET UP PHYSICAL STRUCTURE OF COMPACTED SALE SLIP DATA BASE
      INTEGER*4 TR(11)
      INTEGER*4 NV(5),FL(11,40,45,5),DU(5),COI(200,2),CFI(4000,2,5)
      INTEGER*2 IMT,IDF,IPI
      CHARACTER FI(5)*10,SP*1,YR*2
       INTEGER*2 AI(44),GI(4),PR(39),CO(53)
       LOGICAL*1 DO(3),IFM,JCO(4),IDY,IS2,KCO
      DATA NV/5*0/,COI/400*0/,FL/99000*0/,CFI/40000*0/
      DATA AI/10,21,22,31,32,33,40,56,51,52,
     . 53,54,60,70,80,90,100,110,120,130,
     . 140,150,160,170,180,190,200,210,220,230,
     . 240,250,260,270,280,791,792,793,794,300,
     . 500,400,441,442/
      DATA GI /10,20,30,31/
      DATA PR/31,32,33,34,41,42,43,44,45,51,
     . 52,53,54,61,62,63,64,71,72,73,
     . 74,75,81,82,83,84,91,92,93,94,
     . 101,102,103,104,105,111,112,113,114/
       DATA CO/67,33,50,54,20,51,72,53,52,30
     . ,32,56,57,59,88,11,61,58,80,86,
     . 60,63,64,92,65,66,94,21,68,69,
     . 74,73,75,76,62,77,40,70,82,83,
     . 96,79,84,91,85,87,93,97,78,89,
     . 90,98,99/
 901  FORMAT(' ENTER FILENAME OF SALESLIP DATA: ')
 902  FORMAT(' ENTER YEAR OF DATA (E.G.67): ')
 903  FORMAT(A2)
 904  FORMAT(A10)
 905  FORMAT(2X,2I2,2I3,3X,I5,I2,4X,I5,I2,I7,
     . I8,I1,I7,I2,9X,I5)
       NCO=0
C....PROMPT FOR FILENAMES
C
      WRITE(6,901)
      READ(5,904)FI(1)
  9   WRITE(6,902)
      READ(5,903,ERR=9)YR
      FI(2)='CFV'//YR//'.IDX'
      FI(3)='CO'//YR//'.IDX'
      FI(4)='DB'//YR//'.DAT'
      FI(5)='DB'//YR//'.IDX'
      OPEN(1,FILE=FI(1),STATUS='OLD')
       OPEN(2,FILE=FI(2),STATUS='NEW',ORGANIZATION='RELATIVE',RECL=8,
     . ACCESS='DIRECT',FORM='UNFORMATTED')
C      OPEN(3,FILE=FI(3),STATUS='NEW',ORGANIZATION='RELATIVE',RECL=5,
```

Listing B (cont'd)

```
C     . ACCESS='DIRECT',FORM='UNFORMATTED')
      OPEN(4,FILE=FI(4),STATUS='NEW',ORGANIZATION='RELATIVE',RECL=5,
     . ACCESS='DIRECT',FORM='UNFORMATTED')
      OPEN(5,FILE=FI(5),STATUS='OLD',ORGANIZATION='RELATIVE',RECL=11,
     . ACCESS='DIRECT',FORM='UNFORMATTED')

C
C
C...READ IN DATA BASE INDEX
      IR=0
      DO 1 IGR=1,5
      DO 1 IAR=1,45
      DO 1 IPR=1,40
      IR=IR+1
      READ(5,REC=IR) (FL(I,IPR,IAR,IGR),I=1,11)
    1 CONTINUE
C....READ IN SALESLIP DATA
   10 READ(1,905,END=100,ERR=10)IGR,IDY,IPR,IAR,IDF,
     . ICO,ICFV,ISP,IPS,IWT,IFM,IAMT,IS2,IPI
C....SET UP INDEXES
      CALL INDE(IAR,IPR,IGR,ISP,ICO,AI,GI,PR,CO)
      IR=FL(ISP,IPR,IAR,IGR)
C...SEARCH FOR LAST RECORD INDEX OF SAME CFV
      DO 40 I=NV(IGR),1,-1
   40 IF(CFI(I,1,IGR).EQ.ICFV)GOTO 50
      NV(IGR)=NV(IGR)+1
      I=NV(IGR)
   50 IDX=CFI(I,2,IGR)
      IF(IDX.EQ.0) IDX=-ICFV  !FIRST RECORD=-CFV
      CFI(I,1,IGR)=ICFV
      CFI(I,2,IGR)=IR
C...SEARCH FOR LAST RECORD INDEX OF SAME COMPANY
C     DO 70 I=NCO,1,-1
C  70 IF(COI(I,1).EQ.ICO) GOTO 80
C     NCO=NCO+1
C     I=NCO
C  80 IDC=COI(I,2)
C     IF(IDC.EQ.0) IDC=-ICO  !FIRST RECORD=-COMPANY CODE
C     COI(I,1)=ICO
C     COI(I,2)=IR
C...WRITE OUTPUT FILE
      KCO=ICO
      WRITE(4,REC=IR) IDX,IDF,IPS,IWT,IFM,IPI,KCO,IDY,IS2
      FL(ISP,IPR,IAR,IGR)=FL(ISP,IPR,IAR,IGR)-1
      GOTO 10
C.....WRITE INDEX FILE FOR LAST RECORD OF EACH CFV
```

Listing B (cont'd)

```
  100     IR=1
          DO 60 IGR=1,5
          DO 61 IB=1,4000,4
          IF(CFI(IB,1,IGR).EQ.0)GOTO 62
          IR=IR+1
          IC=IB+3
          WRITE(2,REC=IR) (CFI(I,1,IGR),I=IB,IC),(CFI(I,2,IGR),I=IB,IC)
    61    CONTINUE
    62    DU(IGR)=IR
    60    CONTINUE
          WRITE(2,REC=1) (DU(I),I=1,5)
C... WRITE INDEX FILE FOR LAST RECORD OF EACH COMPANY
C       IR=0
C       DO 85 I=1,200,4
C      IF(COI(I,1).EQ.0) GOTO 90
C       IR=IR+1
C       J=I+3
C       K1=I
C       DO 86 K=1,4
C       JCO(K)=COI(K1,1)
C 86    K1=I+K
C       WRITE(3,REC=IR) (JCO(K),K=1,4),(COI(M,2),M=I,J)
C  85   CONTINUE
   90    STOP
         END
```

Listing C - Listing of report writer program of CDDS (FLEET.FOR)

```
C....PROGRAM TO PRINT CATCH BY PERIOD,AREA,SPECIES OF GIVEN CFV (DB(YR).DAT)
      INTEGER*4 C(6,40,45),DF(40,45),TOT(40,45),SUM(8)
      INTEGER*2 IPI,IDF
      CHARACTER YR*2,Y2*4,GI(5)*10,SP(6)*10,IAS*1,AR*7
      LOGICAL*1 IFM,IDY,IS2,KCO
      DATA GI/'(GILL NET)','(  SEINE )','(  TROLL )','(FR.TROLL)',
     . '(UNKNOWN)'/
      DATA SP/'  SPRING','  SOCKEYE','   COHO','   PINK',
     . '   CHUM','  STEELHEAD'/
      DATA C/10800*0/,DF/1800*0/,TOT/1800*0/,SUM/8*0/
      SW=0
  900 FORMAT(A2)
  901 FORMAT(I5)
  902 FORMAT('1',20X,A4,' CATCH STATISTICS OF SCRAMBLED CFV#: ',I5,A10)
  903 FORMAT(' PERIOD   AREA  ',6A10,'   TOTAL  ','DAYS FISHED')
  904 FORMAT('  TOTAL ',8X,8I10)
  905 FORMAT(A1)
  906 FORMAT(1X,I2,'-',I2,2X,A7,8I10)
  907 FORMAT(1X,I2,'-',I2)
  908 FORMAT(I2)
      OPEN(10,FILE='SCRAT.DAT',STATUS='NEW')
C...PROMPT
   60 PRINT *,' PROGRAM TO PRINT OUT CATCH
     . BY PERIOD,AREA,SPECIES FOR GIVEN CFV#'
      PRINT *
   10 PRINT *,' ENTER YEAR (67<= YEAR <= 80)'
      READ(5,908,ERR=10)IYR
      IF(IYR.GT.80.OR.IYR.LT.67)GOTO 10
      PRINT *
   11 PRINT *,' ENTER SCRAMBLED CFV#: '
      READ(5,901,ERR=11)ICFV
      CALL GFO(IYR,2)
      ENCODE(2,908,YR)IYR
      Y2='19'//YR
C...LOOP OVER GEAR
      DO 1 IGR=1,5
      CALL DBCFV(IGR,ICFV,IDX)
      IF(IDX.EQ.0)GOTO 1
C..SEARCH FOR IAR,IPR,ISP
      CALL DBID1(IGR)
   20 CALL DBID2(IDX,IAR,IPR,ISP)
      READ(13,REC=IDX) INCFV,IDF,IPS,IWT,IFM,IPI,KCO,IDY,IS2
C..OUTPUT MATRIX
      ISP=ISP-4
      IF(ISP.LE.0)ISP=1
```

Listing C (cont'd)

```
      C(ISP,IPR,IAR)=C(ISP,IPR,IAR)+IPS
      DF(IPR,IAR)=DF(IPR,IAR)+IDF
      TOT(IPR,IAR)=TOT(IPR,IAR)+IPS
      IF(INCFV.LE.0)GOTO 30
      IDX=INCFV
      GOTO 20
C....OUTPUT TO 'SCRAT.DAT'
  30  WRITE(10,902)Y2,ICFV,GI(IGR)
      WRITE(10,*)
      WRITE(10,903) (SP(I),I=1,6)
      WRITE(10,*)
      DO 2 IPR=1,40
      DO 3 IAR=1,45
      IF(TOT(IPR,IAR).EQ.0)GOTO 3
       CALL GLIB1(AR,IAR,1)
       CALL GLIB2(YR,IPR,MO,IDAY)
      WRITE(10,906)MO,IDAY,AR,(C(I,IPR,IAR),I=1,6),TOT(IPR,IAR),
     . DF(IPR,IAR)
      DO 4 ISP=1,6
  4   SUM(ISP)=SUM(ISP)+C(ISP,IPR,IAR)
      SUM(7)=SUM(7)+TOT(IPR,IAR)
      SUM(8)=SUM(8)+DF(IPR,IAR)
      SW=1
  3   CONTINUE
      IF(SW.EQ.0)THEN
      CALL GLIB2(YR,IPR,MO,IDAY)
      WRITE(10,907)MO,IDAY
      END IF
      SW=0
  2   CONTINUE
      WRITE(10,*)
      WRITE(10,904) (SUM(J),J=1,8)
C...REINITIALIZE
      SW=0
      DO 5 IAR=1,45
      DO 5 IPR=1,40
      DF(IPR,IAR)=0
      TOT(IPR,IAR)=0
      DO 5 ISP=1,6
      C(ISP,IPR,IAR)=0
  5   CONTINUE
      DO 6 I=1,8
  6   SUM(I)=0
  1   CONTINUE
C...END SESSION
      PRINT *
```

Listing C (cont'd)

```
      PRINT *,' OUTPUT STORED IN FILE 'SCRAT.DAT''
      PRINT *
      PRINT *,' TYPE 'PRINT SCRAT.DAT' TO GET PRINTOUT'
40    PRINT *
      PRINT *,' TRY AGAIN (Y/N)?'
      READ(5,905,ERR=40) IAS
      IF(IAS.EQ.'Y')GOTO 60
      STOP
      END
```

Listing D - Listing of program for generating
CSDS files (DBCAT.FOR)

```
C.....PROGRAM TO SET UP PHYSICAL STRUCTURE OF CATCH DATA BASE
      INTEGER*4 A(3,11,5,40,45),ID(5),BT(5,40,45)
      INTEGER*4 FL(11,40,45,5),TR(11)
      INTEGER*2 GR(5,40,45),TGR(5),DF(5,40,45),IDF,TDF(5)
      CHARACTER FI(5)*10,YR*2
       INTEGER*2 AI(44),GI(4),PR(39)
       LOGICAL*1 DO(3),SP,PDW
      DATA GR/9000*0/,DF/9000*0/,DO/3*0/,BT/9000*0/
      DATA FL/99000*0/
      DATA AI/10,21,22,31,32,33,40,56,51,52,
     . 53,54,60,70,80,90,100,110,120,130,
     . 140,150,160,170,180,190,200,210,220,230,
     . 240,250,260,270,280,791,792,793,794,300,
     . 500,400,441,442/
      DATA GI /10,20,30,31/
      DATA PR/31,32,33,34,41,42,43,44,45,51,
     . 52,53,54,61,62,63,64,71,72,73,
     . 74,75,81,82,83,84,91,92,93,94,
     . 101,102,103,104,105,111,112,113,114/
      DATA A/297000*0/
 901  FORMAT(' ENTER FILENAME OF SALESLIP DATA: ')
 902  FORMAT(' ENTER YEAR OF DATA (E.G.67): ')
 903  FORMAT(A2)
 904  FORMAT(A10)
 905  FORMAT(2X,I2,2X,2I3,3X,I5,6X,I5,I2,I7,I8,I1,7X,I2,14X)
C....PROMPT FOR FILENAMES
C
      WRITE(6,901)
      READ(5,904)FI(1)
  1   WRITE(6,902)
      READ(5,903,ERR=1)YR
      FI(2)='CA'//YR//'.DAT'
      FI(3)='CA'//YR//'.IDX'
      FI(5)='DB'//YR//'.IDX'
      OPEN(1,FILE=FI(1),STATUS='OLD')
      OPEN(2,FILE=FI(2),STATUS='NEW',ORGANIZATION='RELATIVE',RECL=4,
     . ACCESS='DIRECT',FORM='UNFORMATTED')
       OPEN(3,FILE=FI(3),STATUS='NEW',ORGANIZATION='RELATIVE',RECL=11,
     . ACCESS='DIRECT',FORM='UNFORMATTED')

       OPEN(5,FILE=FI(5),STATUS='NEW',ORGANIZATION='RELATIVE',RECL=11,
     . ACCESS='DIRECT',FORM='UNFORMATTED')
C....READ IN SALESLIP DATA
 10   READ(1,905,END=100,ERR=10)
     . IGR,IPR,IAR,IDF,ICFV,ISP,IPS,IWT,IFM,IS2
```

Listing D (cont'd)

```
C....SET UP INDEXES
      IF(IGR.EQ.30.OR.IGR.EQ.31)ISP=IS2
      CALL INDE1(IAR,IPR,IGR,ISP,AI,GI,PR)
C....MATRICES FORMATS
C
C....CATCH MATRIX
C       (#CAUGHT,DRESS WT.,ROUND WT.)
C       BY SPECIES,GEAR,PERIOD,AREA
  12    I=2 !DRESSED WT.
      IF(IFM.EQ.1.OR.IFM.EQ.2)I=3 !ROUND WT.
      A(1,ISP,IGR,IPR,IAR)= A(1,ISP,IGR,IPR,IAR)+IPS
      A(I,ISP,IGR,IPR,IAR)= A(I,ISP,IGR,IPR,IAR)+IWT
       DF(IGR,IPR,IAR)=DF(IGR,IPR,IAR)+IDF
C
C....# GEAR/GEAR/PERIOD/AREA (CFV HAVE TO BE SORTED IN SALESLIP DATA)
      IF(BT(IGR,IPR,IAR).EQ.ICFV)GOTO 11
      GR(IGR,IPR,IAR)=GR(IGR,IPR,IAR)+1
      BT(IGR,IPR,IAR)=ICFV
C...COMPACT SALESLIP DATABASE INDEX
  11  FL(ISP,IPR,IAR,IGR)=FL(ISP,IPR,IAR,IGR)+1
      GOTO 10
C
C....WRITE OUTPUT FILE AND INDEX FILE
C
  100 IRX=0 ! LINK LIST INDEX
      IRN=0 !CATCH DATA INDEX
      DO 140 IAR=1,45
      DO 130 IPR=1,40
      DO 120 IGR=1,5
C....CATCH DATA BASE
      TDF(IGR)=DF(IGR,IPR,IAR)
      TGR(IGR)=GR(IGR,IPR,IAR)
      DO 110 ISP=1,11
      NCA=A(1,ISP,IGR,IPR,IAR)
      IF(NCA.EQ.0)GOTO 110
      IRWT=A(3,ISP,IGR,IPR,IAR)
      IDWT=A(2,ISP,IGR,IPR,IAR)
       ITWT=IRWT+IDWT
       IF(ITWT.EQ.0)THEN
       PDW=0
       GOTO 109
       END IF
       PDW=INT(IDWT/ITWT*100)
 109    SP=ISP
       IRN=IRN+1
      WRITE(2,REC=IRN)SP,NCA,ITWT,PDW
```

Listing D (cont'd)

```
 110  CONTINUE
        ID(IGR)=IRN
 120  CONTINUE
      IRX=IRX+1
      WRITE(3,REC=IRX) (ID(I),I=1,5), (TGR(I),I=1,5)
     . ,(TDF(I),I=1,5),(DO(I),I=1,3)
 130  CONTINUE
 140  CONTINUE
C...COMPACT SALESLIP DATA BAS INDEX
      NR=0
      IR=0
      DO 200 IGR=1,5
      DO 201 IAR=1,45
      DO 202 IPR=1,40
      IR=IR+1
      DO 203 ISP=1,11
      MR=FL(ISP,IPR,IAR,IGR)
      NR=NR+MR
      TR(ISP)=NR
  203 CONTINUE
      WRITE(5,REC=IR) (TR(I),I=1,11)
  202 CONTINUE
  201 CONTINUE
  200 CONTINUE
      STOP
      END
```

Listing E - Listing of report writer program
of CSDS (CATCH.FOR)

```
C...ACCESS PROGRAM TO ONLINE CATCH STAT.
C...FILE EYEAR.DAT STORES LAST YEAR OF CATCH DATA (E.G.80)
      IMPLICIT INTEGER*4(C),CHARACTER*10(X)
      COMMON /CAT/CA(12,41,5),C(11,5)
      COMMON /NAME/XSPI(12),XGRI(3),XFM(2),XPR(40)
      COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      INTEGER*4 IBY(4),TGR(5),TDF(5),DO(3)
      CHARACTER YB*4,YE*4,SNAME*100,IAS*1,YR*2
      CHARACTER*7 AR,SAR(47)
      DATA IBY/51,51,67,67/
      DATA XFM/'PIECES','WEIGHT'/
      DATA XSPI/'  L. RED','  M. RED','   JACK','  WHITE',
     . 'ALL SPRING','  SOCKEYE','   COHO','   PINK',
     . '   CHUM',' STEELHEAD',' DAYS OPEN','DELIVERIES'/
      DATA XGRI/'   ON','  SEINE','  TROLL'/
      DATA XPR/'3-1','3-2','3-3','3-4','4-1','4-2','4-3','4-4','4-5',
     . '5-1','5-2','5-3','5-4','6-1','6-2','6-3','6-4',
     . '7-1','7-2','7-3','7-4','7-5','8-1','8-2','8-3','8-4',
     . '9-1','9-2','9-3','9-4','10-1','10-2','10-3','10-4','10-5',
     . '11-1','11-2','11-3','11-4','D-F'/
  901 FORMAT(I1)
  902 FORMAT(I2)
  903 FORMAT(I3,', ',A10)
  904  FORMAT(' PROCESSING.........')
  905   FORMAT('  ** OPTION 4 NOT IMPLEMENTED ** ')
  906   FORMAT('1 NOTE: 1951-62 WHITE SPRING CATCHES
     ,  ARE TOTAL SPRING CATCHES')
  907   FORMAT('          1963-66 WHITE SPRING TROLL ARE MIXED TROLL')
  908   FORMAT('           'DELIVERIES' ARE 'DAYS FISHING' FOR TROLL')
      OPEN(9,FILE='[SALMON.CATCH]EYEAR.DAT',STATUS='OLD',READONLY)
      OPEN(10,FILE='SCRAT.DAT',STATUS='NEW')
      READ(9,902)IEY
      DO 20 I=1,20
   20 WRITE(10,*)
      WRITE(10,906)
      WRITE(10,907)
      WRITE(10,908)
      CALL INIT
C...PROMPT
   10 IYR=0
      INC=1
      PRINT *,' ACCESS PROGRAM TO ONLINE CATCH STATISTICS'
      PRINT *,'   1. CATCH BY AREA,PERIOD,GEAR,SPECIES'
      PRINT *,'   2. CATCH BY YEAR,AREA,GEAR,SPECIES'
      PRINT *,'   3. CATCH/BOAT WEEK BY AREA,PERIOD,SPECIES'
```

Listing E (cont'd)

```
      PRINT 8,'   4. AVERAGE CATCH/BOAT/DAYS OPEN BY AREA,
     . PERIOD,SPECIES'
      PRINT 8,'   WHICH (1-4)?'
      READ(5,901,ERR=10)IFM
      IF(IFM.LT.1.OR.IFM.GT.4)GOTO 10
      IF(IFM.EQ.4)THEN
      WRITE(6,905)
      GOTO 10
      END IF
   11 PRINT 8,' ENTER STARTING YEAR (>=',IBY(IFM),' )!'
      READ(5,902,ERR=11)IBYR
      IF(IBYR.LT.IBY(IFM).OR.IBYR.GT.IEY)GOTO 11
   12 PRINT 8,' ENTER ENDING YEAR (<= ',IEY,')!'
      READ(5,902,ERR=12)IEYR
      IF(IEYR.LT.51.OR.IEYR.GT.IEY.OR.IEYR.LT.IBYR)GOTO 12
      IF(IFM.EQ.4)THEN
   13 PRINT 8,' ENTER YEAR INCREMENT (1-5)!'
      READ(5,901,ERR=13)INC
      IF(INC.LT.1.OR.INC.GT.5)GOTO 13
      END IF
C.. PROMPT FOR AREA,GEAR,SPECIES
      CALL PRMT1(JSW,ISW,INA,SAR,SNAME)
      IF(IFM.GE.3)THEN
   14 PRINT 8,'  1. GILL NET'
      PRINT 8,'  2. SEINE'
      PRINT 8,'  3. TROLL'
      PRINT 8,'  4. TROLL + FR. TROLL'
      PRINT 8,'  WHICH (1-4)'
      READ(5,901,ERR=14)IBT
      IF(IBT.LT.1.OR.IBT.GT.4)GOTO 14
   15 DO 21 I=1,10
   21 WRITE(6,903)I,XSPI(I)
      PRINT 8,' 11. SPECIES 5 TO 9 INCLUSIVE'
      PRINT 8,' WHICH (1-11)?'
      READ(5,901,ERR=15)ISP1
      IF(ISP1.LT.1.OR.ISP1.GT.11)GOTO 15
      END IF
C
C...LOOP OVER YEAR
C
      PRINT 8
      WRITE(6,904)
      JA=1
      IA=1
    6 IAR=JAR(IA,SAR)
      DO 1 IY=IEYR,IBYR,-INC
```

Listing E (cont'd)

```
      IYR=IYR+1
      ENCODE(2,902,YR)IY
      CALL GFO(IY,1)
C..LOOP OVER AREA
  5   IAR=JAR(IA,SAR)
      AR=SAR(IA)
C..LOOP OVER PERIOD,GEAR
      IF(IFM.EQ.2)THEN
       IPR=41
       CALL CAA1(IAR,IPR,ISW,C,TGR,TDF,DO)
       CALL OM1(IYR,DO,TDF)
      ELSE
      DO 3 IPR=1,40
      CALL CAA1(IAR,IPR,ISW,C,TGR,TDF,DO)
C..OUTPUT MATRIX,(CA)
      IF(JSW.EQ.1)JA=JA+1
      IF(IFM.EQ.1)CALL OM1(IPR,DO,TDF)
      IF(IFM.EQ.3)CALL OM3(IYR,TGR,IPR)
      IF(IFM.EQ.4)CALL OM3(JA,TGR,IPR)
   3  CONTINUE
       END IF
C
C..OUTPUT FORMATS
      IF(IFM.EQ.1.AND.JSW.EQ.1)CALL OF1(SNAME,AR,YR)
      IF(IFM.GE.2.AND.JSW.EQ.1)GOTO 1
      IA=IA+1
      IF(IA.LE.INA)GOTO 5
      IA=1
       IF(IFM.EQ.1.AND.JSW.EQ.2)CALL OF1(SNAME,AR,YR)

    1  CONTINUE
      IF(IFM.EQ.1)GOTO 90
      IF(IFM.EQ.2)CALL OF2(SNAME,AR)
      IF(IFM.EQ.3)CALL OF3(SNAME,AR)
      IF(IFM.NE.4.AND.JSW.EQ.2)GOTO 90
      IA=IA+1
      IYR=0
      IF(IA.LE.INA)GOTO 6
      IF(IFM.EQ.4)CALL OF4(SNAME,SAR,INC,INA)
   90   PRINT 8,' TRY AGAIN (Y/N)?'
        READ(5,910,ERR=90)IAS
        IF(IAS.EQ.'Y')GOTO 10
        PRINT 8
        PRINT8,' OUTPUT STORED IN FILE 'SCRAT.DAT''
        PRINT 8
        PRINT 8,' TYPE 'PRINT SCRAT.DAT' TO GET PRINTOUT'
```

Listing E (cont'd)

```
      PRINT *
910   FORMAT(A1)
      STOP
      END
     SUBROUTINE INIT
      IMPLICIT INTEGER*4(C)
      COMMON /CAT/CA(12,41,3),C(11,5)
     DO 1 I=1,3
     DO 1 J=1,41
     DO 1 K=1,12
     CA(K,J,I)=0
1    CONTINUE
     RETURN
     END
      FUNCTION JAR(IA,SAR)
      CHARACTER*7 SAR(47),AR
      AR=SAR(IA)
      CALL GLIB1(AR,IAR,0)
      JAR=IAR
      RETURN
      END
```

Listing E (cont'd)

```
C...SUBROUTINE TO PROMPT FOR AREA COMBINATIONS
C   JSW=1:SINGLE AREA; JSW=2:SUM; ISW=1,2:PIECES,WEIGHT; INA:# AREAS; SAR:STAT.AREAS WANTED
      SUBROUTINE PRMT1(JSW,ISW,INA,SAR,SNAME)
      INTEGER*2 NA(5)
      CHARACTER SNAME*100,SNA(5)*40
      CHARACTER*7 SAR(47),AR1(55),AR,S1
      DATA AR1/'1','2E','2W','3','4','5','6',
     . '7','8','9','10','30','11','12','13',
     . '14','15','16','17','18','19','20','21','22','23',
     . '24','25','26','27','28','C','29AB','29C','29D','29E',
     . 'STIKINE','TAKU','ALASKA','UNKNOWN',
     . '3X','3Y','3Z','5/1','5/2','5/3','5/4','5/6',
     . '2AE','2BE','2AW','2BW','D1','D2','D3','BC'/
      DATA NA/0,55,53,54,52/
      DATA SNA/'0','TOTAL B.C.','NORTH COAST SUMMARY','SOUTH COAST
     .SUMMARY','FRASER AREA SUMMARY'/
  900 FORMAT(' ENTER THE ',I2,' AREAS:')
  901 FORMAT(' AREA # ',I2)
  902 FORMAT(10A8)
  903  FORMAT(' ERROR!!  REENTER AREA !!')
  800 FORMAT(I1)
  801 FORMAT(I2)
  802 FORMAT(A7)
  803 FORMAT(A100)
C...PROMPT
   10  PRINT *
       PRINT *,' 1. CATCH BY PIECES'
       PRINT *,' 2. CATCH BY WEIGHT'
       PRINT *,' WHICH (1 OR 2) ?'
       READ(5,800,ERR=10) ISW
       IF(ISW.LT.1.OR.ISW.GT.2)GOTO 10
C
   11  PRINT *
       PRINT *,'WANT TO SUM MORE THAN ONE AREA (Y/N)?'
       READ(5,802,ERR=11)S1
       IF(S1.NE.'Y'.AND.S1.NE.'N')GOTO 11
       IF(S1.EQ.'N')JSW=1
       IF(S1.EQ.'N')GOTO 13
       JSW=2
   12  PRINT *
       PRINT *,' 1. ANY COMBINATION OF AREAS OR SUBAREAS'
       PRINT *,' 2. TOTAL B.C.'
       PRINT *,' 3. NORTH COAST SUMMARY (STAT.AREA 1-10,30)'
       PRINT *,' 4. SOUTH COAST SUMMARY (STAT.AREA 11-27,C)'
       PRINT *,' 5. FRASER AREA SUMMARY (STAT.AREA 28+29)'
```

Listing E (cont'd)

```
      PRINT *,' WHICH (1-5)?'
      READ(5,800,ERR=12) IAS
      IF(IAS.LT.1.OR.IAS.GT.5)GOTO 12
      IF(IAS.GT.1) THEN
      INA=1
      SNAME=SNA(IAS)
      RETURN
      END IF
   13 PRINT *
      PRINT *,'AREAS AVAILABLE:'
      WRITE(6,902) (AR1(I),I=1,10)
      WRITE(6,902)(AR1(I),I=11,20)
      WRITE(6,902)(AR1(I),I=21,30)
      WRITE(6,902)(AR1(I),I=31,40)
      WRITE(6,902)(AR1(I),I=41,47)
      PRINT *
      IF(JSW.EQ.1)PRINT *,'ENTER # OF AREAS OR SUBAREAS WANTED
     . (1-44 OR ENTER 99 TO PRINT ALL AREAS):'
      IF(JSW.EQ.2)PRINT *,'ENTER # OF AREAS OR SUBAREAS TO SUM (2-47):'
      READ(5,801,ERR=13)INA
      IF(INA.EQ.99)THEN
      INA=47
      DO 2 I=1,47
    2 SAR(I)=AR1(I)
      RETURN
      END IF
      IF(INA.LT.1.OR.INA.GT.45)GOTO 13
   14 WRITE(6,900)INA
      DO 3 I=1,INA
   15 WRITE(6,901) I
      READ(5,802,ERR=15)SAR(I)
C....ERROR CHECKING
       DO 16 J=1,47
   16   IF(SAR(I).EQ.AR1(J))GOTO 3
        WRITE(6,903)
        GOTO 15
    3  CONTINUE
       IF(JSW.EQ.1)RETURN
   70  PRINT *,'ENTER NAME OF AREA COMBINATION (E.G. GEORGIA STRAIT)'
       READ(5,803,ERR=70) SNAME
       RETURN
       END
```

Listing E, (cont'd)

```
C...SUBROUTINE TO FORM OUTPUT MATRIX (CA) FORMAT 1 AND 2
      SUBROUTINE OM1(IYR,DO,TDF)
      IMPLICIT INTEGER*4(C)
       COMMON /CAT/  CA(12,41,5),C(11,5)
       COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      INTEGER*4 TDF(5),DO(3)
      DO 1 IGR=1,4
      IG=IGR
      IF(IG.EQ.4)IG=3 !FR.TROLL+TROLL
C
C..FIRST PAGE
      DO 2 ISP=1,5
      IS=ISP
      IF(ISP.GE.4)IS=IS-1
      IO=(IS-1)*3+IG
      CA(IO,IYR,1)=CA(IO,IYR,1)+C(ISP,IGR)
      CA(IO,41,1)=CA(IO,41,1)+C(ISP,IGR)
  2   CONTINUE
C
C..SECOND PAGE
      DO 3 ISP=1,8
      IS=ISP-4
      IF(IS.LT.1)IS=1
      IO=(IS-1)*3+IG
      CA(IO,IYR,2)=CA(IO,IYR,2)+C(ISP,IGR)
      CA(IO,41,2)=CA(IO,41,2)+C(ISP,IGR)
  3   CONTINUE
C
C..THIRD PAGE
      DO 4 ISP=9,10
      IS=ISP-8
      IO=(IS-1)*3+IG
      CA(IO,IYR,3)=CA(IO,IYR,3)+C(ISP,IGR)
      CA(IO,41,3)=CA(IO,41,3)+C(ISP,IGR)
   4  CONTINUE
C
C..DAYS OPEN AND TOTAL DAYS FISHED
      IF(IGR.EQ.4)GOTO 10
      IO=IG+6
      CA(IO,IYR,3)=CA(IO,IYR,3)+DO(IGR)
      CA(IO,41,3)=CA(IO,41,3)+DO(IGR)
 10   IO=IG+9
      CA(IO,IYR,3)=CA(IO,IYR,3)+TDF(IGR)
      CA(IO,41,3)=CA(IO,41,3)+TDF(IGR)
  1   CONTINUE
```

Listing E (cont'd)

```
RETURN
END
```

Listing E (cont'd)

```
C...SUBROUTINE TO FORM OUTPUT MATRIX FORMAT 3 AND 4
      SUBROUTINE OM3(IYR,TGR,IPR)
      IMPLICIT INTEGER*4 (C)
      COMMON /CAT/CA(12,41,5),C(11,5)
      COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      INTEGER*4 TGR(5)
      DO 3 I=1,5
    3 IF(TGR(I).EQ.0)TGR(I)=1
      NP=1
      IS=ISP1
      IF(IS.EQ.4)IS=5 !WHITE SPRING
      IF(ISP1.EQ.3)THEN
      DO 1 I=3,4
    1 CA(IYR,IPR,N)=CA(IYR,IPR,N)+C(I,IBT)
      GOTO 11
      END IF
      IF(ISP1.EQ.11.OR.ISP1.EQ.5)THEN
      DO 2 I=1,5
    2 CA(IYR,IPR,NP)=CA(IYR,IPR,NP)+C(I,IBT)
      IS=5
      GOTO 11
      END IF
   10 CA(IYR,IPR,NP)=CA(IYR,IPR,NP)+C(IS,IBT)
   11 IF(ISP1.EQ.11)THEN
      IS=IS+1
      NP=NP+1
      IF(IS.LE.9)GOTO 10
      END IF
      RETURN
      END
```

Listing E (cont'd)

```
C...SUBROUTINE TO PRINT FORM 1
      SUBROUTINE OF1(SNAME,AR,YR)
      IMPLICIT INTEGER*4(C),CHARACTER*10(X)
       COMMON /CAT/  CA(12,41,5),C(11,5)
       COMMON /NAME/  XSPI(12),XGRI(3),XFM(2),XPR(40)
      COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      CHARACTER SNAME*100,Y2*4,YR*2,AR*7
  900 FORMAT('1',40X,'REVISED CATCH STATISTICS BY AREA  IN ',A7,A4)
  903 FORMAT('   AREA : ',A7)
  906 FORMAT(8X,4(7X,A10,7X))
  907 FORMAT(' WK. ENDING ',12A8)
  908 FORMAT(1X,I2,'-',I2,2X,12I8)
  909 FORMAT('  TOTAL  ',12I8)
  910  FORMAT('  D-F ',2X,12I8)
  912 FORMAT('  AREA : ',A100)
      Y2='19'//YR
C..LOOP OVER PAGES
      DO 1 NP=1,3
      WRITE(10,900)XFM(ISW),Y2
      WRITE(10,*)
      IF(JSW.EQ.1)WRITE(10,903)AR
      IF(JSW.EQ.2)WRITE(10,912)SNAME
      WRITE(10,*)
      J=(NP-1)*4+1
      K=J+3
      WRITE(10,906) (XSPI(I),I=J,K)
      WRITE(10,*)
      WRITE(10,907) ((XGRI(I),I=1,3),J=1,4)
      WRITE(10,*)
      DO 5 IPR=1,39
      CALL GLIB2(YR,IPR,MO,IDAY)
  5   WRITE(10,908)MO,IDAY,(CA(J,IPR,NP),J=1,12)
       WRITE(10,910)(CA(J,40,NP),J=1,12)
      WRITE(10,*)
      WRITE(10,909)(CA(J,41,NP),J=1,12)
  1   CONTINUE
      CALL INIT
      RETURN
      END
```

Listing E (cont'd)

```
C...SUBROUTINE TO PRINT FORM 2
      SUBROUTINE OF2(SNAME,AR)
      IMPLICIT INTEGER*4(C),CHARACTER*10(X)
      COMMON /CAT/CA(12,41,5),C(11,5)
      COMMON /NAME/XSPI(12),XGRI(3),XFM(2),XPR(40)
      COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      CHARACTER SNAME*100,AR*7
  900 FORMAT('1',40X,'CATCH BY SPECIS AND GEAR  IN ',A7,
     , '19',I2,'-19',I2)
  901 FORMAT('   AREA : ',A7)
  902 FORMAT('   YEAR   ',12A8)
  903 FORMAT(3X,'19',I2,3X,12I8)
  904 FORMAT('   AREA : ',A100)
  906 FORMAT(8X,4(7X,A10,7X))
C...LOOP OVER PAGES
      DO 1 NP=1,3
      JY=0
      WRITE(10,900)XFM(ISW),IBYR,IEYR
      WRITE(10,*)
      IF(JSW.EQ.1)WRITE(10,901)AR
      IF(JSW.EQ.2)WRITE(10,904)SNAME
      WRITE(10,*)
      J=(NP-1)*4+1
      K=J+3
      WRITE(10,906)(XSPI(I),I=J,K)
      WRITE(10,*)
      WRITE(10,902)((XGRI(I),I=1,3),J=1,4)
      WRITE(10,*)
      DO 5 IY=IEYR,IBYR,-1
      JY=JY+1
 5    WRITE(10,903)IY,(CA(J,JY,NP),J=1,12)
 1    CONTINUE
      CALL INIT
      RETURN
      END
```

Listing E (cont'd)

```
C...SUBROUTINE TO PRINT FORMAT 3
      SUBROUTINE OF3(SNAME,AR)
      IMPLICIT INTEGER*4 (C),CHARACTER*10(X)
      COMMON /CAT/CA(12,41,5),C(11,5)
      COMMON /NAME/XSPI(12),XGRI(3),XFM(2),XPR(40)
      COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      CHARACTER SNAME*100,AR*7
      INTEGER*2 IY(20)
  900 FORMAT('1',20X,A12,'CATCH/',A10,'BOAT WEEK  IN ',
     . A7,'  19',I2,'-19',I2)
  901 FORMAT('  AREA : ',A7)
  902 FORMAT('  PERIOD  ',12I8)
  903 FORMAT(2X,A6,2X,12I8)
  904 FORMAT('  AREA : ',A100)
      NY=0
      DO 1 I=IEYR,IBYR,-1
      NY=NY+1
    1 IY(NY)=I
C...LOOP OVER PAGES
      NP=1
      IS=ISP1
      IF(IS.EQ.11)IS=5
   10 WRITE(10,900)XSPI(IS),XGRI(IBT),XFM(ISW),IBYR,IEYR
      WRITE(10,*)
      IF(JSW.EQ.1)WRITE(10,901)AR
      IF(JSW.EQ.2)WRITE(10,904)SNAME
      WRITE(10,*)
      WRITE(10,902) (IY(I),I=1,NY)
      WRITE(10,*)
      DO 2 IPR=1,40
    2 WRITE(10,903)XPR(IPR),(CA(I,IPR,NP),I=1,NY)
      IF(ISP1.NE.11)GOTO 90
      IS=IS+1
      NP=NP+1
      IF(NP.LE.5)GOTO 10
      CALL INIT
   90 RETURN
      END
```

Listing E (cont'd)

```
C...SUBROUTINE TO PRINT FORMAT 4
      SUBROUTINE OF4(SNAME,SAR,INC,INA)
      IMPLICIT INTEGER*4 (C),CHARACTER*10 (X)
      COMMON /CAT/CA(12,41,5),C(11,5)
      COMMON /NAME/XSPI(12),XGRI(3),XFM(2),XPR(40)
      COMMON INTEGER /IDX/ISP1,IBT,ISW,JSW,IBYR,IEYR
      CHARACTER SNAME*100,AR*7,SAR(45)
  900 FORMAT('1',20X,'AVERAGE',A7,'CATCH/',A7,
     . 'BOAT WEEK  IN ',A7,'  19',I2,'-19',I2,' STEP',I2)
  901 FORMAT(40X,'STATISTICAL AREA')
  902 FORMAT('  PERIOD  ',12A8)
  903 FORMAT(2X,A6,2X,12I8)
  904 FORMAT('  PERIOD  ',A100)
      NY=0
      DO 1 I=IEYR,IBYR,-INC
    1 NY=NY+1
C...LOOP OVER PAGES
      NP=1
      IS=ISP1
      IF(IS.EQ.11)IS=5
   10 WRITE(10,900) XSPI(IS),XGRI(IBT),XFM(ISW),IBYR,
     . IEYR,INC
      WRITE(10,*)
      WRITE(10,901)
      WRITE(10,*)
      IF(JSW.EQ.1)WRITE(10,902)(SAR(I),I=1,INA)
      IF(JSW.EQ.2)WRITE(10,904)SNAME
      IF(PR.EQ.2)INA=1
      WRITE(10,*)
      DO 2 IPR=1,40
      DO 2 I=1,INA
    2 CA(I,IPR,NP)=CA(I,IPR,NP)/NY
      DO 3 IPR=1,40
    3 WRITE(10,903)XPR(IPR),(CA(I,IPR,NP),I=1,INA)
      IF(ISP1.NE.11)GOTO 90
      IS=IS+1
      NP=NP+1
      IF(NP.LE.5)GOTO 10
   90 CALL INIT
      RETURN
      END
```

Listing F - Listing of subroutines to access
CSDS and CDDS

```
C...SUBROUTINE TO READ IN CATCH C(11 SPECIES,5 GR), BGR TGR(5),DAYS FISHED
C   TDF(5/GR),DAYS OPEN(3/GR)- BY AREA,PERIOD IN PIECES IF ISW=1,WT. IF ISW=2
      SUBROUTINE CAA1(IAR,IPR,ISW,C,TGR,TDF,DO)
      INTEGER*4 C(11,5),ID(5),CA(11,3),TGR(5),TDF(5),DO(3)
      IF(ISW.NE.1)ISW=2
      CALL CAGR(IAR,IPR,ID,TGR,TDF,DO)
      DO 1 IGR=1,5
      CALL CAREC(IAR,IPR,IGR,ID,CA)
      DO 2 ISP=1,11
    2 C(ISP,IGR)=CA(ISP,ISW)
    1 CONTINUE
      RETURN
      END
```

Listing F (cont'd)

```
C....SUBROUTINE TO READ IN INDEX AND TGR(#GEAR/GEAR),TDF(TOT.DAYS FISHED),
C    DO(DAYS OPEN/GEAR)
      SUBROUTINE CAGR(IAR,IPR,ID,TGR,TDF,DO)
      INTEGER*4 ID(5),TGR(5),TDF(5),DO(3)
      INTEGER*2 JGR(5),JDF(5)
      LOGICAL*1 JDO(3)
      MX=1
      IF(IPR.EQ.41.OR.IAR.GT.51)MX=10   !TOTALS STORED IN 10S
      IR=(IAR-1)*41+IPR
      READ(12,REC=IR) (ID(I),I=1,5),(JGR(I),I=1,5),
     . (JDF(I),I=1,5),(JDO(I),I=1,3)
      DO 1 I=1,5
      TGR(I)=JGR(I)*MX
 1    TDF(I)=JDF(I)*MX
      DO 2 I=1,3
 2    DO(I)=JDO(I)*MX
      RETURN
      END
```

Listing F (cont'd)

```
C....SUBROUTINE TO READ IN CA(BY SPECIES,#CAUGHT,TOT.WT.,%RD.WT.),
      SUBROUTINE CAREC(IAR,IPR,IGR,ID,CA)
      INTEGER*4 ID(5),CA(11,3),DU1(4)
      INTEGER*2 DU2(5)
      LOGICAL*1 SP,PDW,DU3(3)
      DO 11 I=1,3
      DO 11 J=1,11
      CA(J,I)=0
 11   CONTINUE
      MX=1
      IF(IPR.EQ.41.OR.IAR.GT.51)MX=10
      IR=(IAR-1)*41+IPR
      IF(IGR.NE.1)IJ=ID(IGR-1)+1
      IF(IGR.NE.1)GOTO 20
      IF(IR.EQ.1)IJ=1
      IF(IR.EQ.1)GOTO 20
      IR2=IR-1
      READ(12,REC=IR2) (DU1(I),I=1,4),IJ,(DU2(I),I=1,5),
     . (DU2(I),I=1,5),(DU3(I),I=1,3)
      IJ=IJ+1
 20   DO 1 IRC=IJ,ID(IGR)
      READ(11,REC=IRC) SP,NCA,ITWT,PDW
      I=SP
      CA(I,1)=NCA*MX
      CA(I,2)=ITWT*MX
      CA(I,3)=PDW
 1    CONTINUE
      RETURN
      END
```

Listing F (cont'd)

```
C....SUBROUTINE TO SEARCH FOR LAST ADDRESS OF GIVEN CFV IN CFV(YR).IDX
      SUBROUTINE DBCFV(IGR,ICFV,IDX)
      INTEGER*4 ID(5),CFV(4),RC(4)
      READ(16,REC=1)(ID(I),I=1,5)
      IB=2
      IF(IGR.NE.1)IB=ID(IGR-1)+1
      DO 1 I=IB,ID(IGR)
      READ(16,REC=I)(CFV(J),J=1,4),(RC(J),J=1,4)
      DO 2 IC=1,4
      IF(ICFV.EQ.CFV(IC))THEN
      IDX=RC(IC)
      RETURN
      END IF
    2 CONTINUE
    1 CONTINUE
      IDX=0
      RETURN
      END
*
```

Listing F (cont'd)

```
C...SUBROUTINE TO RETURN CATCH BY AREA,PERIOD,SPECIES GIVEN CFV#
      SUBROUTINE DBCFVD(IGR,IYR,ICFV,ISW,C,DF)
      INTEGER*4 C(11,40,45),DF(40,45)
      INTEGER*2 IPI,IDF
      LOGICAL*1 IFM,IDY,IS2,KCO
C...ERROR CHECK
      I=0
      IF(IGR.GT.5.OR.IGR.LT.1)I=1
      IF(I.NE.0)CALL ERR(I)
C
      DO 1 I=45
      DO 1 J=1,40
      DO 2 K=1,11
      C(K,J,I)=0
  2   CONTINUE
      DF(J,I)=0
  1   CONTINUE
      IDX=0
C     CALL GFO(IYR,2)
      CALL DBCFV(IGR,ICFV,IDX)
      IF(IDX.EQ.0)RETURN
C..SEARCH FOR IPR,IAR,ISP
      CALL DBID1(IGR)
  20  CALL DBID2(IDX,IAR,IPR,ISP)
      READ(13,REC=IDX)INCFV,IDF,IPS,IWT,IFM,IPI,KCO,IDY,IS2
      IF(ISW.NE.1)IPS=IWT
      C(ISP,IPR,IAR)=C(ISP,IPR,IAR)+IPS
      DF(IPR,IAR)=DF(IPR,IAR)+IDF
      IF(INCFV.LE.0)RETURN
      IDX=INCFV
      GOTO 20
      END
```

Listing F (cont'd)

```
C....SUBROUTINE TO SEARCH FOR IAR,IPR,ISP,GIVEN ADDRESS OF DB(YR).DAT
      SUBROUTINE DBID1(IGR)
      INTEGER*4 ID(45,40,11)
      SAVE ID
C..READ IN RECORD INDEX FROM DB(YR).IDX
      IR=(IGR-1)*1800
      DO 1 IA=1,45
      DO 1 IP=1,40
      IR=IR+1
  1   READ(14,REC=IR) (ID(IA,IP,IS),IS=1,11)
      RETURN
C...SEARCH FOR IAR,IPR,ISP
      ENTRY DBID2(IDX,IAR,IPR,ISP)
      IE=ID(45,40,11)
      DO 2 IA=44,1,-1
      IB=ID(IA,40,11)
      IF(IDX.GT.IB.AND.IDX.LE.IE)THEN
      IAR=IA+1
      GOTO 11
      END IF
      IE=IB
   2  CONTINUE
      IAR=1
C...SEARCH FOR PERIOD
  11  IE=ID(IAR,40,11)
      DO 3 IP=39,1,-1
      IB=ID(IAR,IP,11)
      IF(IDX.GT.IB.AND.IDX.LE.IE)THEN
      IPR=IP+1
      GOTO 21
      END IF
      IE=IB
  3   CONTINUE
      IPR=1
C...SEARCH FOR SPECIES
  21  IE=ID(IAR,IPR,11)
      DO 4 IS=10,1,-1
      IB=ID(IAR,IPR,IS)
      IF(IDX.GT.IB.AND.IDX.LE.IE)THEN
      ISP=IS+1
      GOTO 31
      END IF
      IE=IB
  4   CONTINUE
      ISP=1
```

Listing F (cont'd)

```
31  RETURN
    END
```

Listing F (cont'd)

```
C...SUBROUTINE TO RETURN STARTING AND ENDING ADDRESS, GIVEN GEAR, AREA, PERIOD,
C.. SPECIES
      SUBROUTINE DBIDX(IGR,IAR,IPR,ISP,IS,IE)
      INTEGER ID(11)
C.. ERROR CHECK
      I=0
      IF(IGR.LT.1.OR.IGR.GT.5)I=1
      IF(IAR.LT.1.OR.IAR.GT.45)I=2
      IF(IPR.LT.1.OR.IPR.GT.40)I=3
      IF(ISP.LT.1.OR.ISP.GT.11)I=4
      IF(I.NE.0)CALL ERR(I)
C..FIND ADDRESS
      IR=(IGR-1)*1800+(IAR-1)*40+IPR
      READ(14,REC=IR) (ID(I),I=1,11)
      IE=ID(ISP)
      IF(ISP.NE.1)THEN
      IS=ID(ISP-1)+1
      RETURN
      END IF
      IS=1
      IF(IR.EQ.1)RETURN
      IR=IR-1
      READ(14,REC=IR) (IDU,I=1,10),IS
      IS=IS+1
      RETURN
      END
```

Listing F (cont'd)

```
C....SUBROUTINE TO OPEN APPROPIATE DATA AND INDEX FILES
      SUBROUTINE GFO(IYR,ISW)
      CHARACTER FI(6)*30,YR*2,DE*19
  902 FORMAT(I2)
      ENCODE(2,902,YR)IYR
      DE='DRA1:[SALMON.CATCH]'
      FI(1)=DE//'CA'//YR//'.DAT'
      FI(2)=DE//'CA'//YR//'.IDX'
      FI(3)=DE//'DB'//YR//'.DAT'
      FI(4)=DE//'DB'//YR//'.IDX'
      FI(5)=DE//'CD'//YR//'.IDX'
      FI(6)=DE//'CFV'//YR//'.IDX'
      OPEN(11,FILE=FI(1),STATUS='OLD',ORGANIZATION='RELATIVE',
     . RECL=4,ACCESS='DIRECT',FORM='UNFORMATTED',READONLY)
      OPEN(12,FILE=FI(2),STATUS='OLD',ORGANIZATION='RELATIVE',
     . RECL=11,ACCESS='DIRECT',FORM='UNFORMATTED',READONLY)
      IF(ISW.EQ.1)GOTO 90
      OPEN(13,FILE=FI(3),STATUS='OLD',ORGANIZATION='RELATIVE',
     . RECL=5,ACCESS='DIRECT',FORM='UNFORMATTED',READONLY)
      OPEN(14,FILE=FI(4),STATUS='OLD',ORGANIZATION='RELATIVE',
     . RECL=11,ACCESS='DIRECT',FORM='UNFORMATTED',READONLY)
C     OPEN(15,FILE=FI(5),STATUS='OLD',ORGANIZATION='RELATIVE',
C    . RECL=5,ACCESS='DIRECT',FORM='UNFORMATTED',READONLY)
      OPEN(16,FILE=FI(6),STATUS='OLD',ORGANIZATION='RELATIVE',
     . RECL=8,ACCESS='DIRECT',FORM='UNFORMATTED',READONLY)
   90 RETURN
      END
```

**Listing F (cont'd)**

```
C...LIBRARY SUBROUTINE TO CONVERT STAT. AREAS TO DATABASE AREAS
      SUBROUTINE GLIB1(AR,IAR,IW)
      CHARACTER STAR(55)*7,AR*7
      DATA STAR/'1','2E','2W','3X','3Y','3Z','4','5/6','5/1',
     . '5/2','5/3','5/4',
     . '6','7','8','9','10','11','12','13','14','15',
     . '16','17','18','19','20','21','22','23','24','25',
     . '26','27','28','29AB','29C','29D','29E','30','C','ALASKA','TAKU',
     . 'STIKINE','UNKNOWN','2AE','2BE','2AW','2BW','3','5',
     . 'D1','D2','D3','BC'/
      IF(IW.EQ.1)THEN
      AR=STAR(IAR)
      RETURN
      END IF
      DO 1 I=1,55
    1 IF(AR.EQ.STAR(I))GOTO 10
      IAR=0 !NO SUCH CODE
      RETURN
   10 IAR=I
      RETURN
      END
```

Listing F (cont'd)

```
C...SUBROUTINE TO RETURN ENDING DATE OF GIVEN PERIOD,YEAR
C.. DY STORES ENDING DATE OF 1ST WK OF MAY (PERIOD 10 IN YEAR YR1)
      SUBROUTINE GLIB2(YR,IPR,MO,IDAY)
      INTEGER DY(36),MON(12)
      CHARACTER YR*2,YR1(36)*2
      DATA DY/12,10,9,8,7,12,11,10,9,7,13,12,11,9,8,7,
     . 13,11,10,9,8,6,5,11,10,8,7,13,12,10,6*0/
      DATA YR1/'51','52','53','54','55','56','57','58',
     . '59','60','61','62','63','64','65','66',
     . '67','68','69','70','71',
     . '72','73','74','75','76','77','78',
     . '79','80',6*'0'/
      DATA MON/31,28,31,30,31,30,31,31,30,31,30,31/
      DO 1 IY=1,36
    1 IF(YR.EQ.YR1(IY))GOTO 10
      IDAY=0
      MO=0
      RETURN
   10 MO=5
      IDY=(IPR-10)*7
      IDAY=DY(IY)+IDY
      IF(IDAY.LE.0)GOTO 30
   20 IF(IDAY.GT.MON(MO))THEN
      IDAY=IDAY-MON(MO)
      MO=MO+1
      GOTO 20
      END IF
      RETURN
   30 MO=MO-1
      IDAY=IDAY+MON(MO)
      IF(IDAY.LE.0)GOTO 30
      RETURN
      END
```